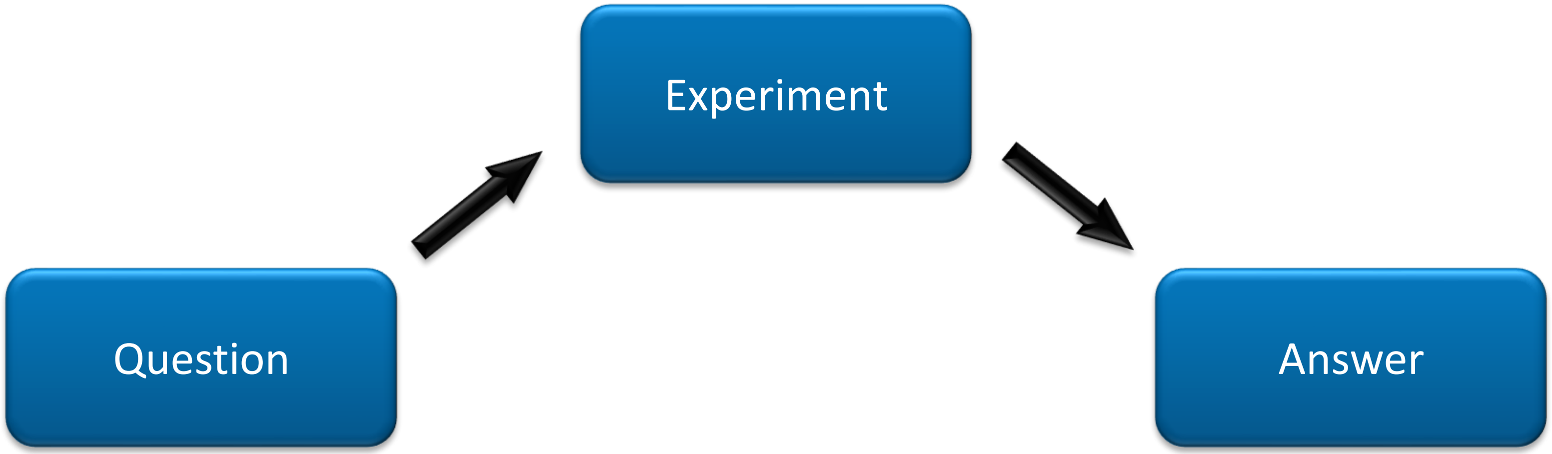
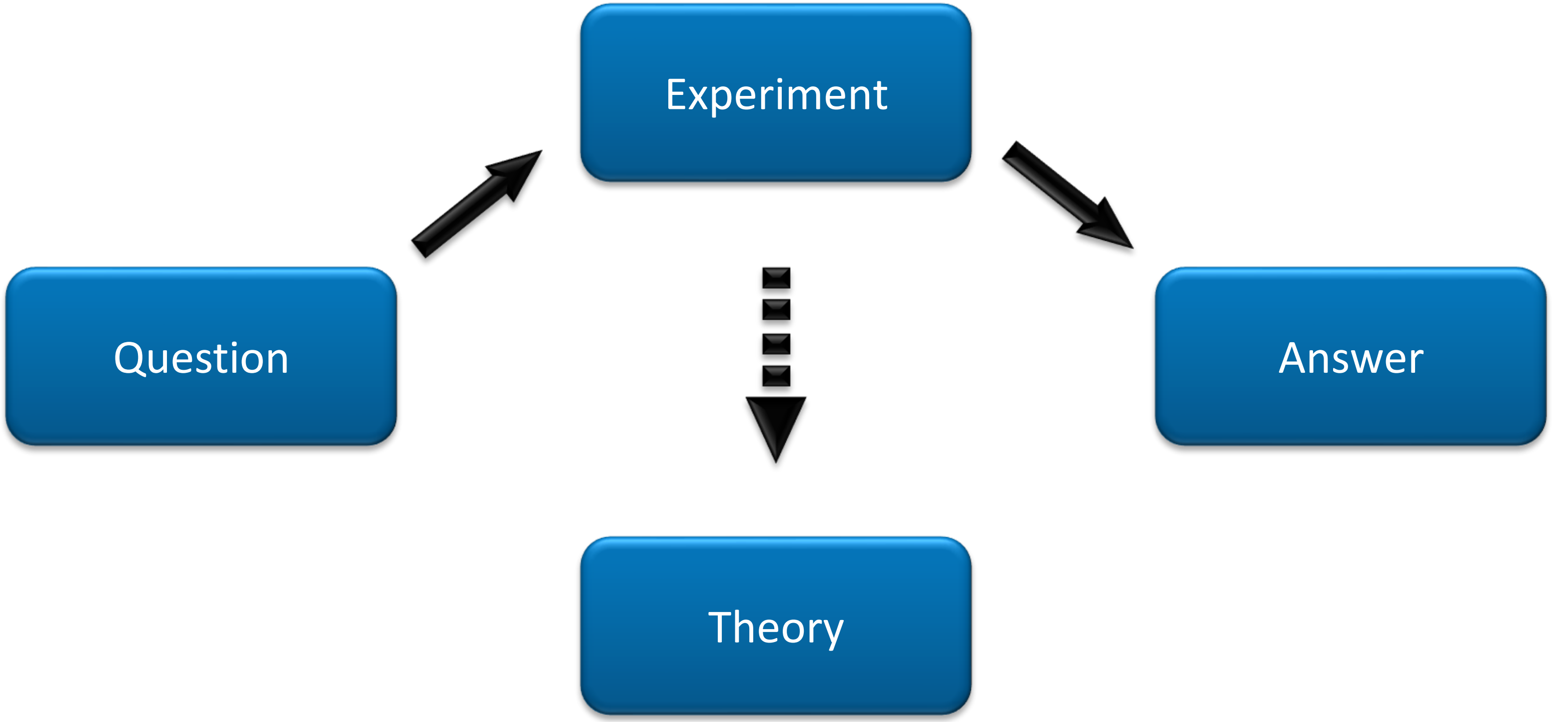
Modelling and Simulation

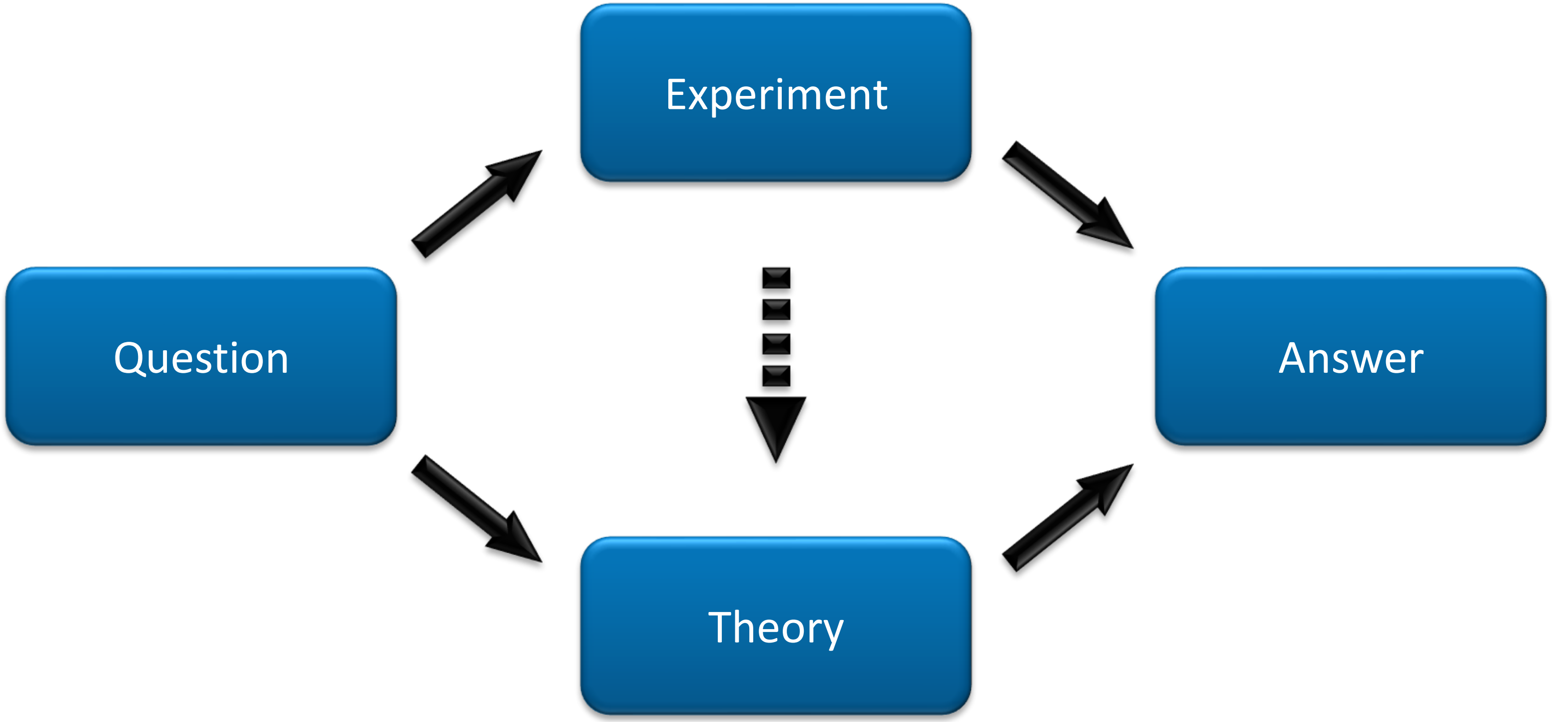
Third Pillar of Science


Question
?
Answer

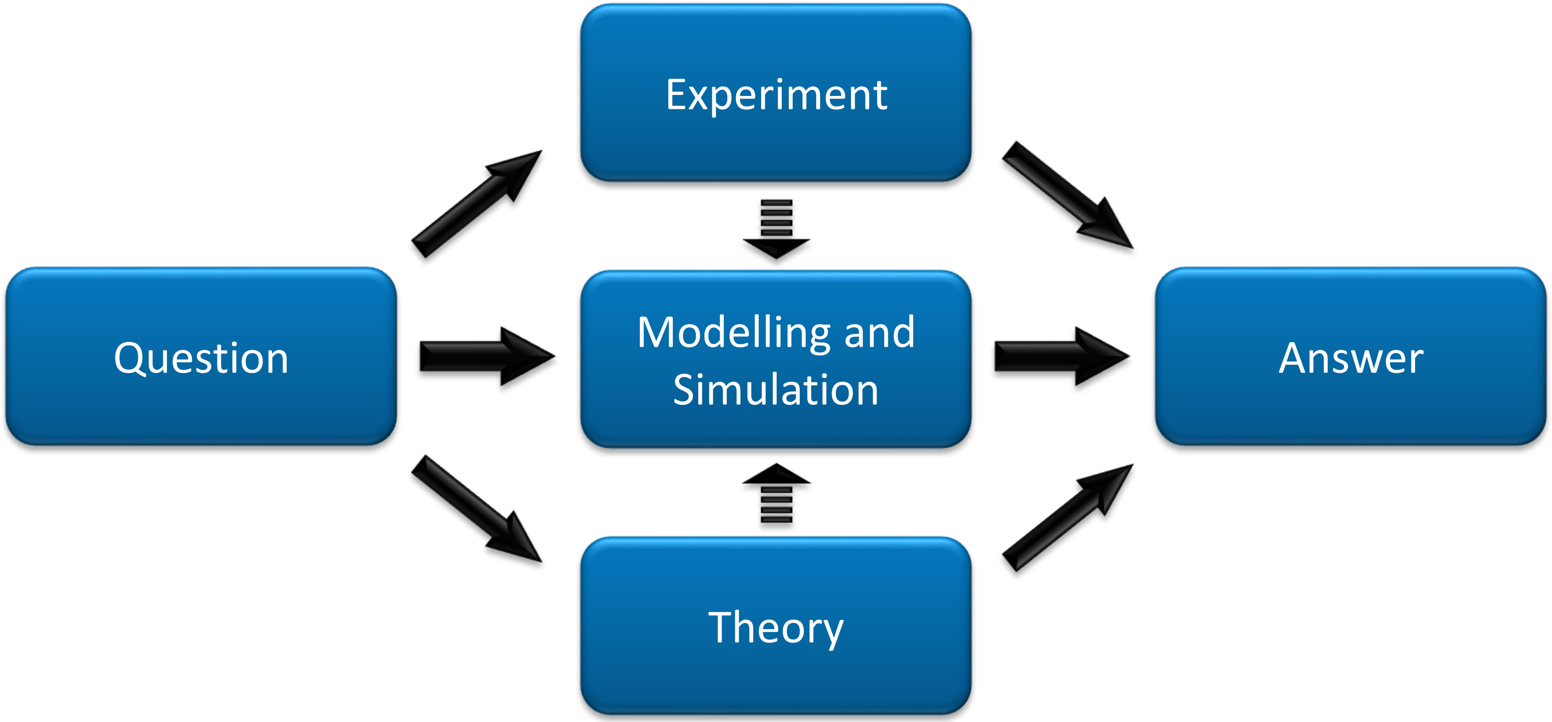

Experiment
Question
Answer


Experiment
Question
Answer
Theory

Mathematical Modelling and Simulation

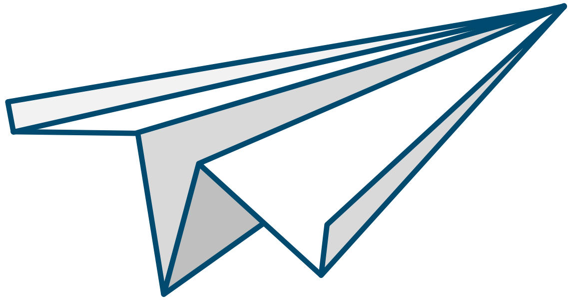


Mathematical Modelling and Simulation



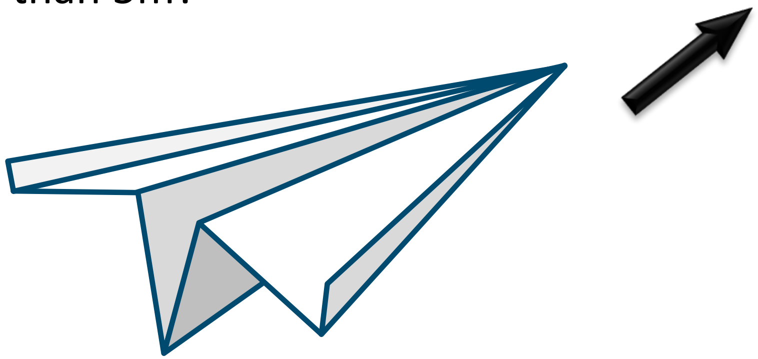
Abstract Example

Does the paper plane fly further than 5m?

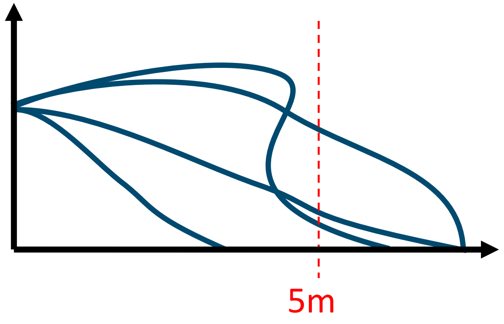


Mathematical Modelling and Simulation
Abstract Example

Does the paper plane fly further than 5m?

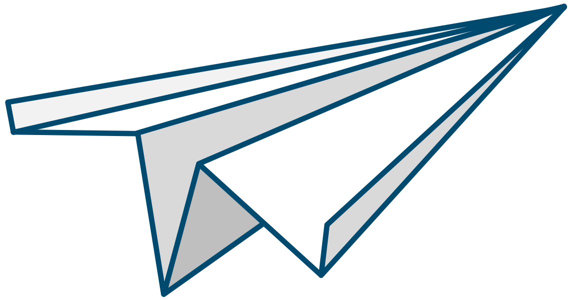


Experiments

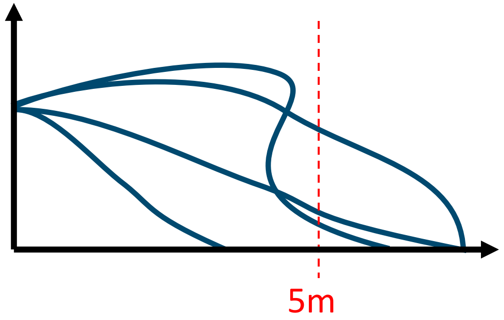


Abstract Example

Does the paper plane fly further than 5m?



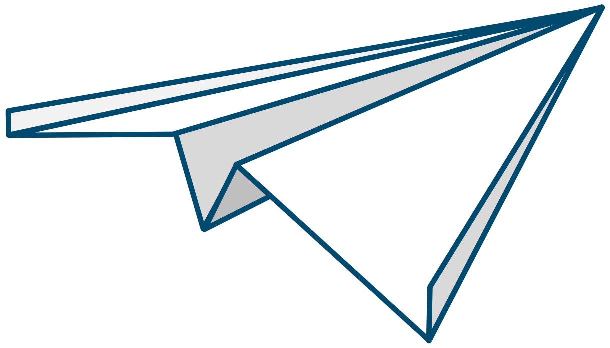
Experiments



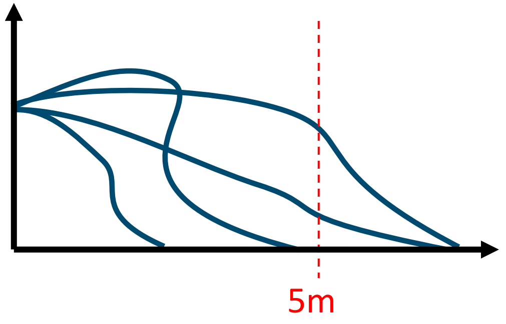
In 80% of all cases.

Abstract Example

Does the paper plane fly further than 5m?



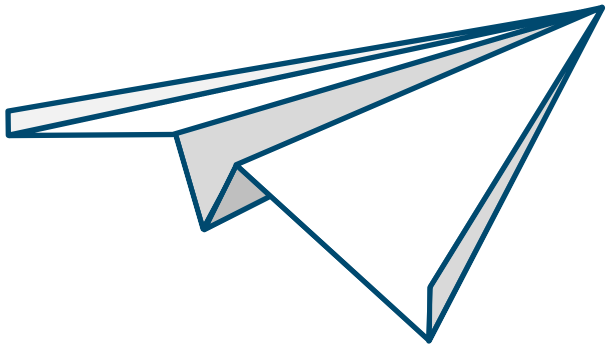
Experiments



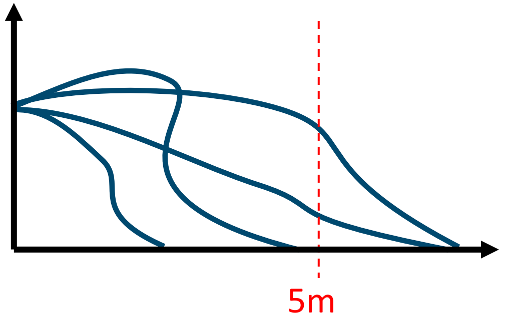
In 70% of all cases.

Mathematical Modelling and Simulation
Abstract Example

Does the paper plane fly further than 5m?



Experiments

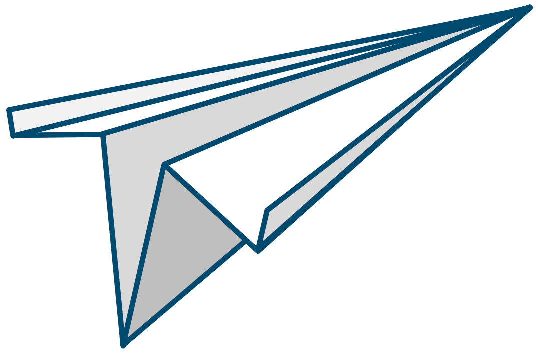


In 70% of all cases.

General Paper Plane Theory:
„Arrow-type paper planes fly ... dependent on their wing span...“

Abstract Example

Does the paper plane fly further than 5m?



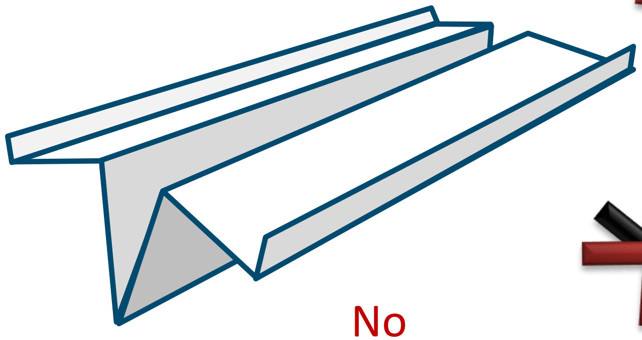
General Paper Plane Theory:
„Arrow-type paper planes fly ... dependent on their wing span...“

In 90% of all cases.

Mathematical Modelling and Simulation
Abstract Example

I need to know it in advance

Does the paper plane fly further than 5m?



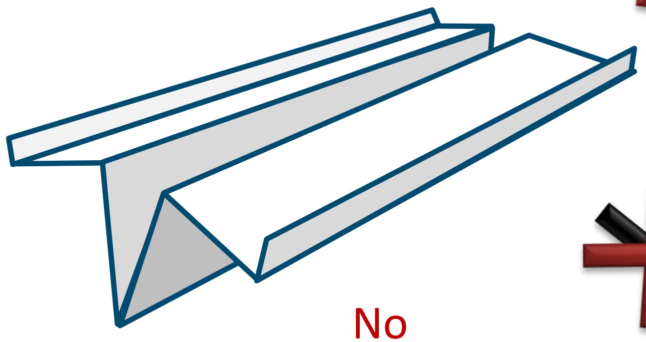
No
arrow-type

General Paper Plane Theory:
„Arrow-type paper planes fly ... dependent on their wing span...“

Mathematical Modelling and Simulation
Abstract Example

I need to know it in advance

Does the paper plane fly further than 5m?



No arrow-type

Mathematical paper plane model

$$e = \frac{u^2}{2} + \frac{p}{\rho} + gz$$

- Findings from „ General Paper Plane Theory“
- Data from experiments
- Additional knowledge like Physikal laws, ...

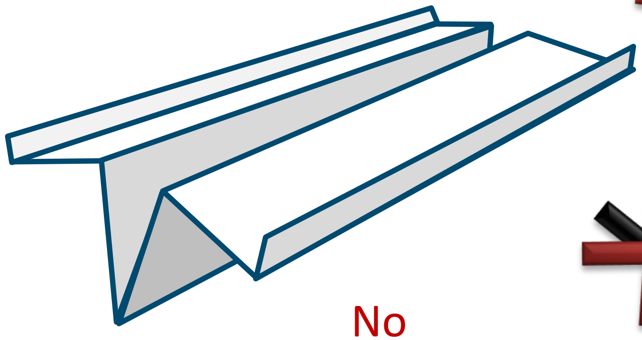
General Paper Plane Theory:
„Arrow-type paper planes fly ... dependent on their wing span...“

Mathematical Modelling and Simulation
Abstract Example

Mathematical Modelling and Simulation

I need to know it in advance

Does the paper plane fly further than 5m?



No arrow-type

Mathematical paper plane model

$$e = \frac{u^2}{2} + \frac{p}{\rho} + gz$$

Simulation

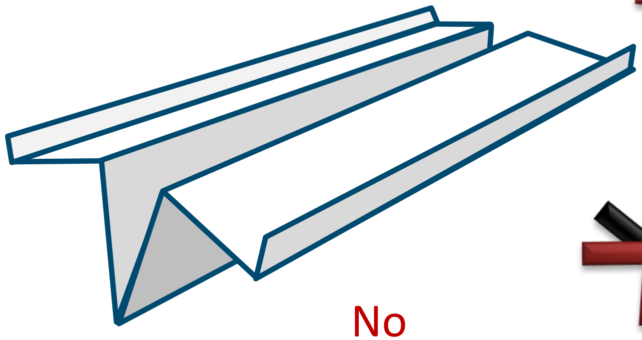
In 2% of all cases.

General Paper Plane Theory:
„Arrow-type paper planes fly ... dependent on their wing span...“

Mathematical Modelling and Simulation
Abstract Example

I need to know it in advance

Does the paper plane fly further than 5m?



No arrow-type

Mathematical paper plane model

$$e = \frac{u^2}{2} + \frac{p}{\rho} + gz$$

Simulation

In 2% of all cases... probably

General Paper Plane Theory:
„Arrow-type paper planes fly ... dependent on their wing span...“

Model Uncertainty:

- Previous knowledge cannot be used directly
- Impossible to model all influence factors
- ...

How is SARS-CoV-2 going to spread in Austria?
What consequences will the epidemic of COVID-19 have on the population?

- Italy ≠ Austria

Unintentional „experiments“: data from countries in which the virus has started spreading earlier

How is SARS-CoV-2 going to spread in Austria?
What consequences will the epidemic of COVID-19 have on the population?

- SARS CoV-2 ≠ Influenza

General knowledge about Influenza, other corona viruses, and the health care system

Mathematical Modelling and Simulation COVID-19

- Italy ≠ Austria

Unintentional „experiments“: data from countries in which the virus has started spreading earlier

How is SARS-CoV-2 going to spread in Austria?
What consequences will the epidemic of COVID-19 have on the population?

COVID-19 simulation model

e.g.:

$$S' = -\alpha I S$$
$$I' = \alpha I S - \beta I$$
$$R' = \beta I$$

- Information / data from countries with more „COVID-19 experience“
- General knowledge about the spread of infectious diseases in Austria
- Additional causal knowledge about the spread of infectious diseases (SIR)

- SARS CoV-2 ≠ Influenza

General knowledge about Influenza, other corona viruses, and the health care system

Unintentional „experiments“: data from countries in which the virus has started spreading earlier

- Italy ≠ Austria

How is SARS-CoV-2 going to spread in Austria?
What consequences will the epidemic of COVID-19 have on the population?

COVID-19 simulation model

e.g.:

$$S' = -\alpha IS$$
$$I' = \alpha IS - \beta I$$
$$R' = \beta I$$

- Information / data from countries with more „COVID-19 experience“
- General knowledge about the spread of infectious diseases in Austria
- Additional causal knowledge about the spread of infectious diseases (SIR)

Model Uncertainty ?

General knowledge about Influenza, other corona viruses, and the health care system

- SARS CoV-2 ≠ Influenza

Mathematical Modelling and Simulation

Introduction to Modelling and Simulation

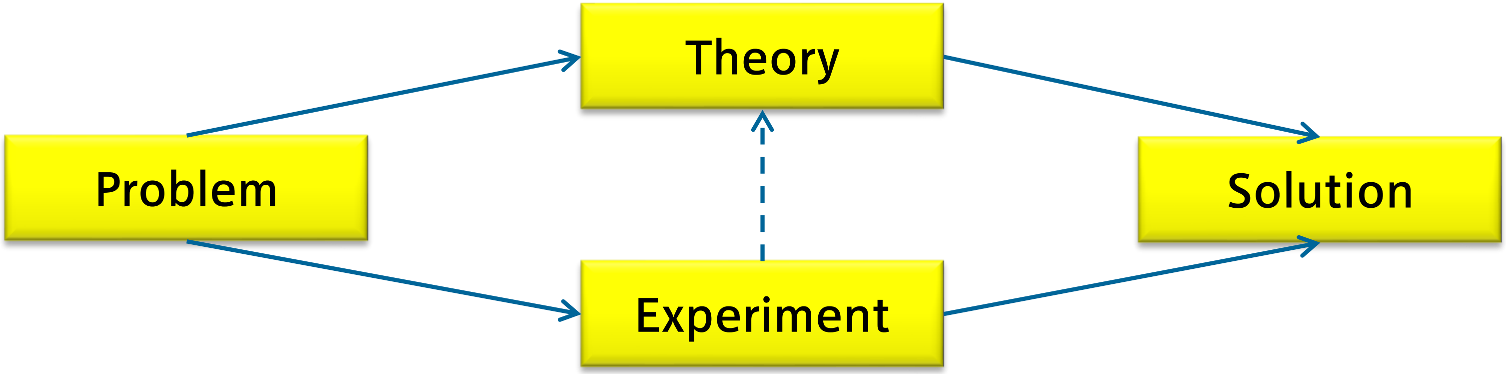
Methods and Algorithms

Classical Approach

Classical Scientific Problem
- Application of Theories
- Execute Experiments

Classical Approach

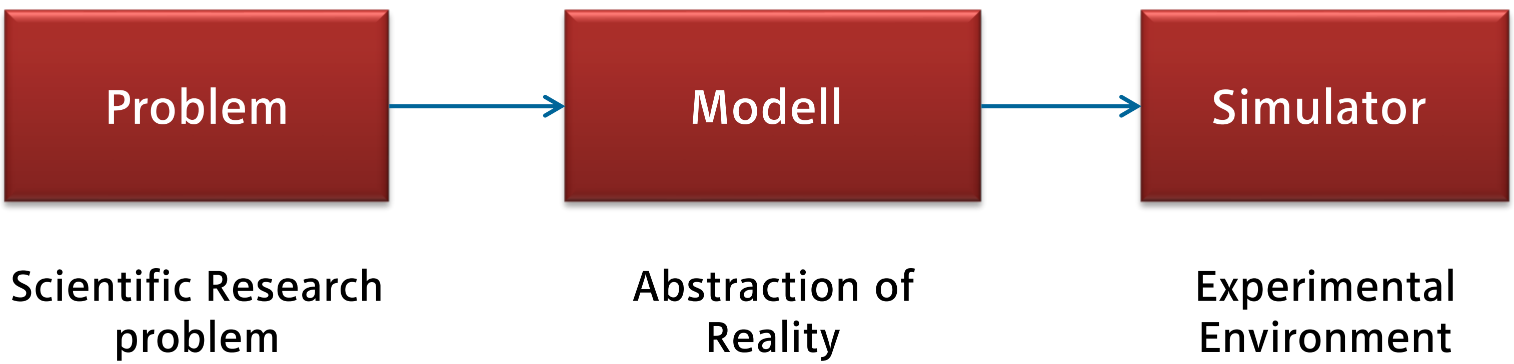
Classical Scientific Problem
- Application of Theories
- Execute Experiments



Simulation

Simulation

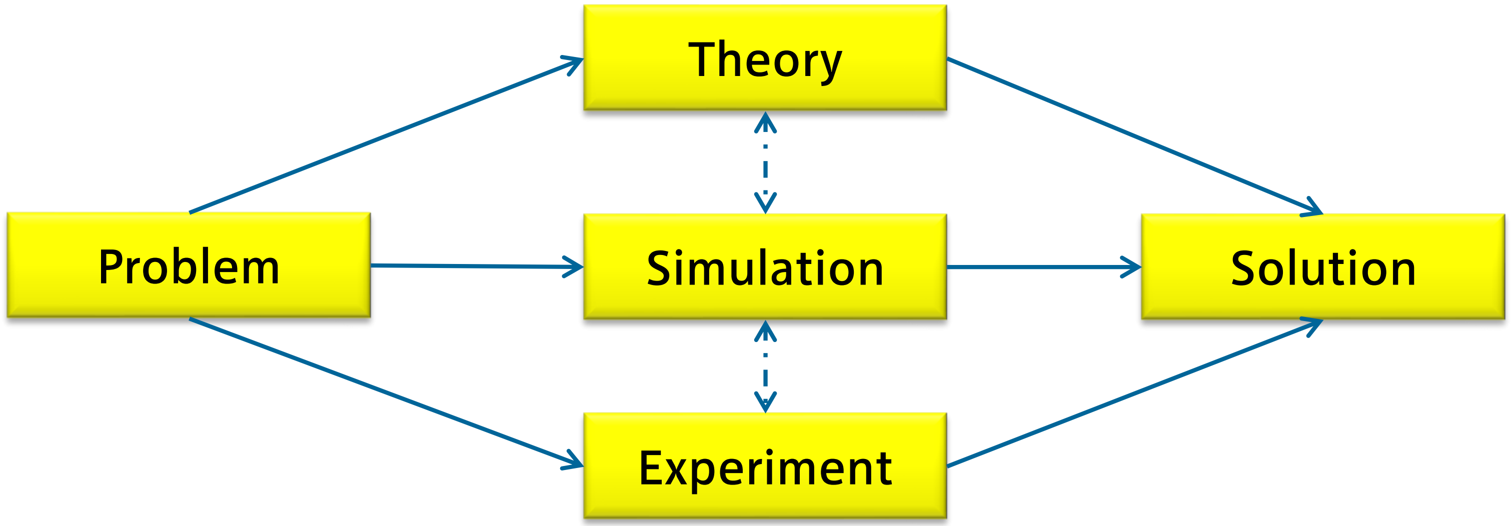
- Experiments in virtual laboratory
- Experiments in the computer
- The third pillar of science beside theory and experiment



Simulation

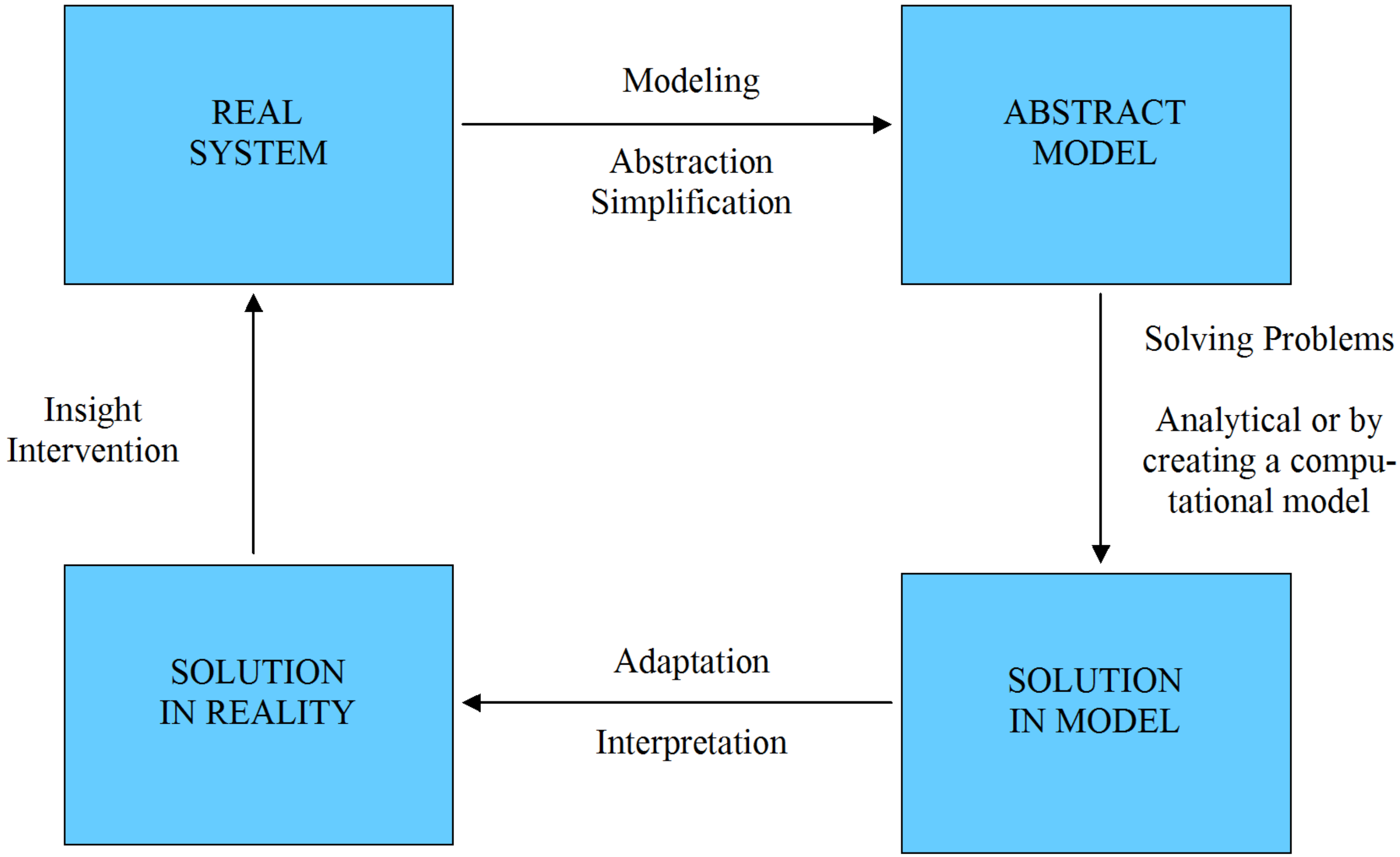
Mathematical Modelling and Simulation

Simulation
- Experiments in virtual laboratory
- Experiments in the computer
- The third pillar of science beside theory and experiment



Solution finding

Mathematical Modelling and Simulation



What is Computer Simulation ?

Mathematical Modelling and Simulation

Definition (Shannon, 1975)

Simulation is the process of designing a model of a real system and conducting experiments with this model for the purpose either of understanding the behavior of the system and its underlying causes or of evaluating various designs of an artificial system or strategies for the operation of the system.

What is Computer Simulation ?

Mathematical Modelling and Simulation

Definition 2 (VDI-Richtlinie 3633)

Simulation is a (virtual) copy of a real system with its dynamic processes in a (virtual) model (computer model) and (virtual) experiments with experiments with this model, which allow interpretations for the real system.

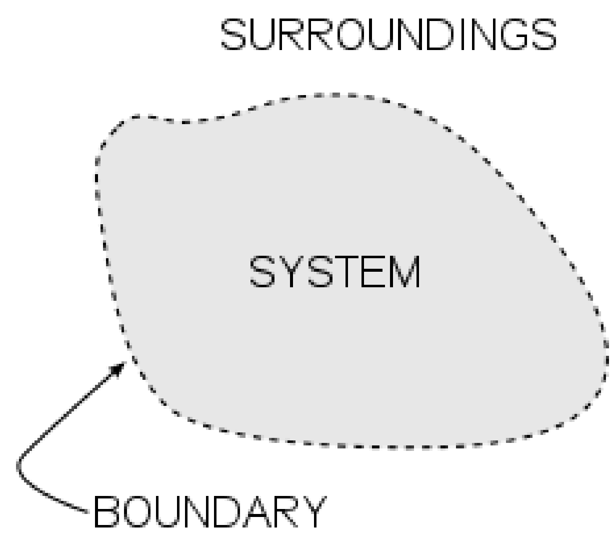
In a practical sense, simulation is i) preparing, ii) performing, and iii) evaluating experiments with a simulation model.

Simulation allows to study time-dependent behaviour of complex dynamical systems in a simulation model.

Dynamical Systems

A system is a set of interacting or interdependent components forming an integrated whole

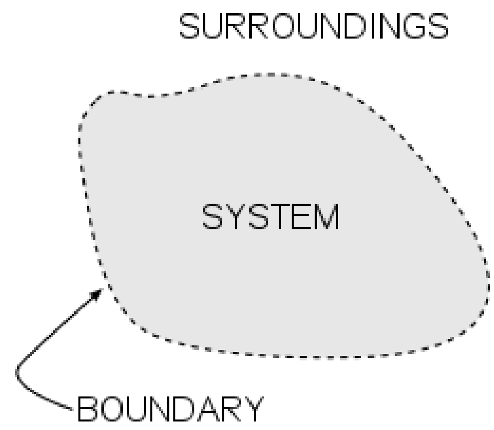
A dynamic system is a set of dynamically interacting or interdependent components forming an integrated whole



Dynamical Systems

Mathematical Modelling and Simulation

A dynamic system is a set of dynamically interacting or interdependent components forming an integrated whole

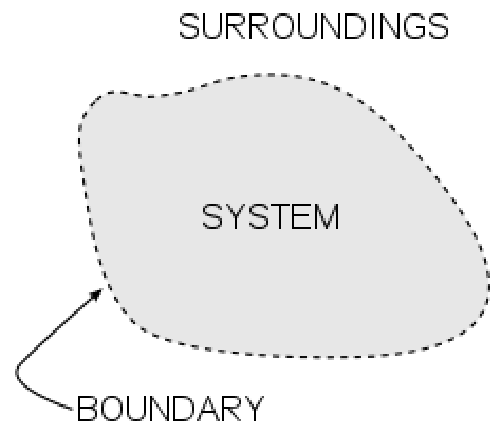


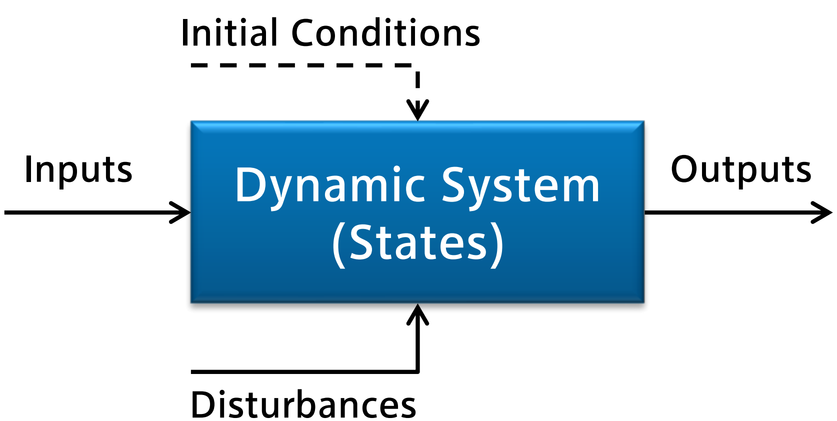
- Dynamical systems change their behaviour dependent on acting input signals, disturbances, and initial values
-
- The behaviour of a dynamical system is not direct proportional to input and disturbance change, it changes its behaviour on basis of its own dynamic and on inputs.

Dynamical Systems

Mathematical Modelling and Simulation

- Dynamical systems change their behaviour dependent on acting input signals, disturbances, and initial values
-
- The behaviour of a dynamical system is not direct proportional to input and disturbance change, it changes its behaviour on basis of its own dynamic and on inputs.



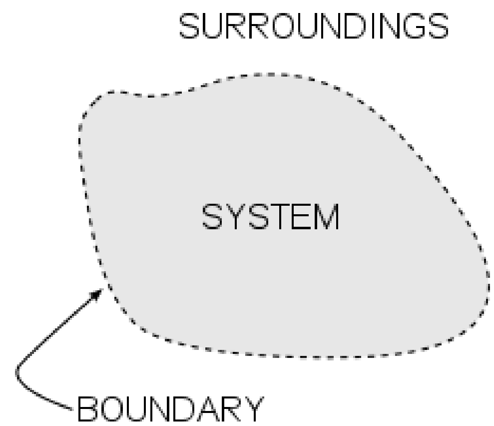


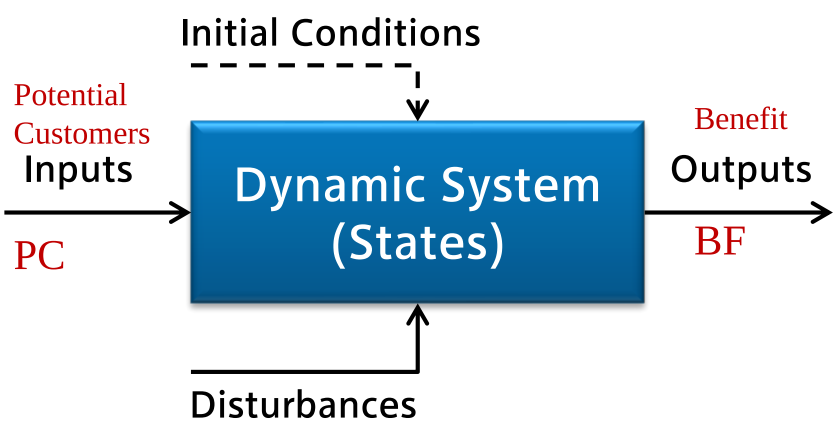
Elements of a Dynamical System

- States **x**(t)
- Inputs **u**(t)
- Disturbances **w**(t) = Inputs
- Outputs **y**(t)
- Fixed Parameters, Intial Conditions
- Time dependent Parameters (Inputs)

Dynamical Systems

- Dynamical systems change their behaviour dependent on acting input signals, disturbances, and initial values
-
- The behaviour of a dynamical system is not direct proportional to input and disturbance change, it changes its behaviour on basis of its own dynamic and on inputs.



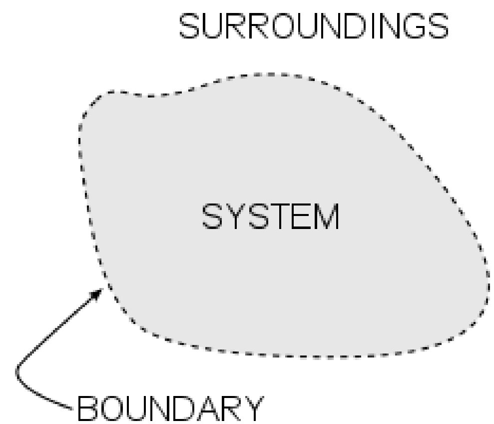


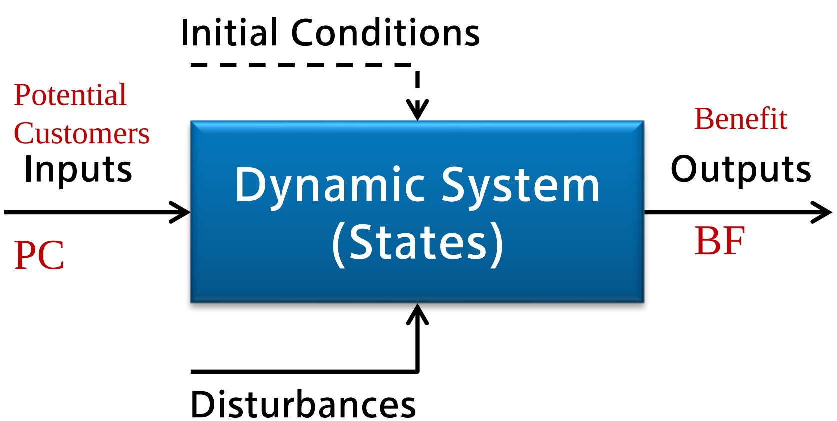
Elements of a Dynamical System

- States **x**(t)
- Inputs **u**(t)
- Disturbances **w**(t) = Inputs
- Outputs **y**(t)
- Fixed Parameters, Intial Conditions
- Time dependent Parameters (Inputs)

Dynamical Systems

- Dynamical systems change their behaviour dependent on acting input signals, disturbances, and initial values
-
- The behaviour of a dynamical system is not direct proportional to input and disturbance change, it changes its behaviour on basis of its own dynamic and on inputs.



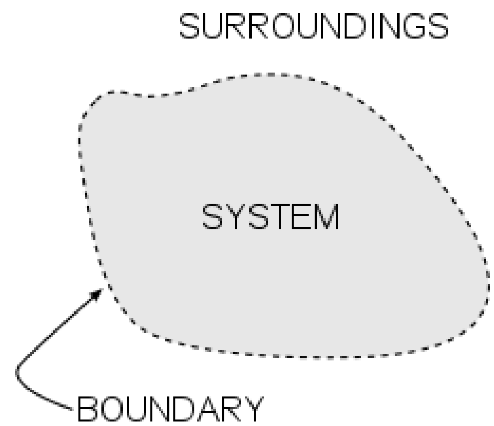


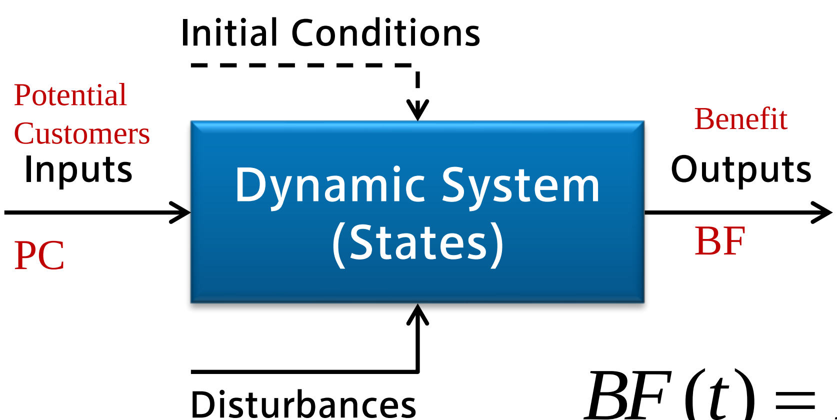
static

$$BF = Faktor \cdot PC$$

Dynamical Systems

- Dynamical systems change their behaviour dependent on acting input signals, disturbances, and initial values
-
- The behaviour of a dynamical system is not direct proportional to input and disturbance change, it changes its behaviour on basis of its own dynamic and on inputs.





static

$$BF = Faktor \cdot PC$$

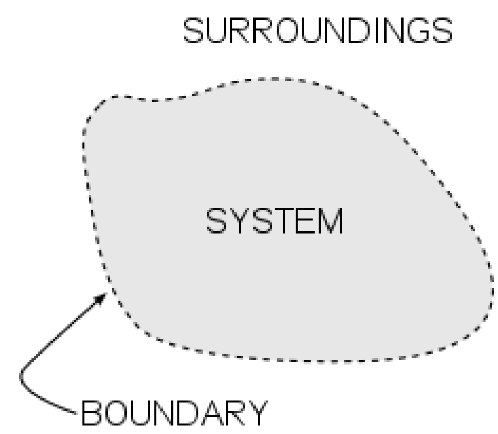
dynamic

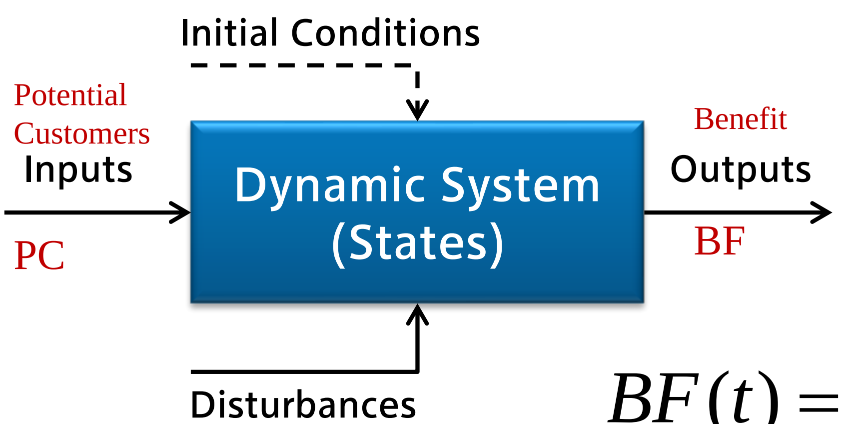
$$BF(t) = Function(PC(t), t, Parameters)$$

Dynamical Systems

- Dynamical systems change their behaviour dependent on acting input signals, disturbances, and initial values

- The behaviour of a dynamical system is not direct proportional to input and disturbance change, it changes its behaviour on basis of its own dynamic and on inputs.





static

$$BF = Faktor \cdot PC$$

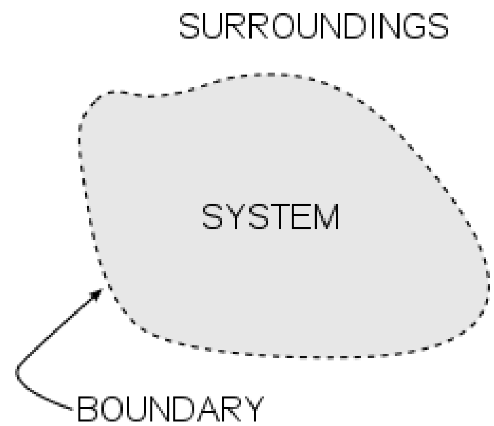
really dynamic

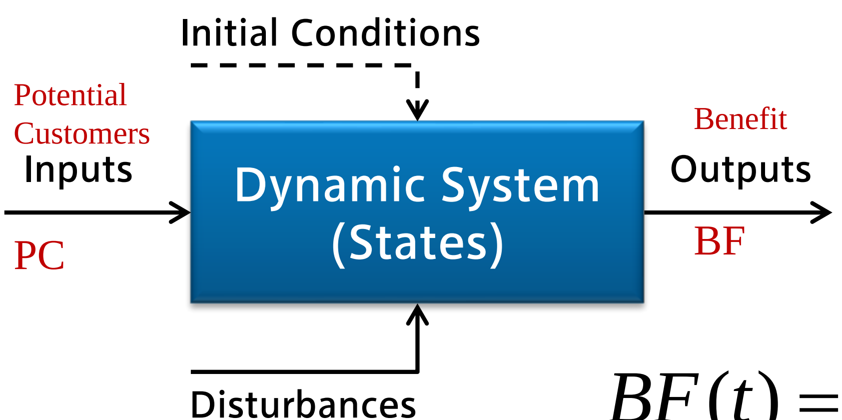
$$BF(t) = Function(PC(t), BF(t), t, Par)$$

Dynamical Systems

- Dynamical systems change their behaviour dependent on acting input signals, disturbances, and initial values

- The behaviour of a dynamical system is not direct proportional to input and disturbance change, it changes its behaviour on basis of its own dynamic and on inputs.





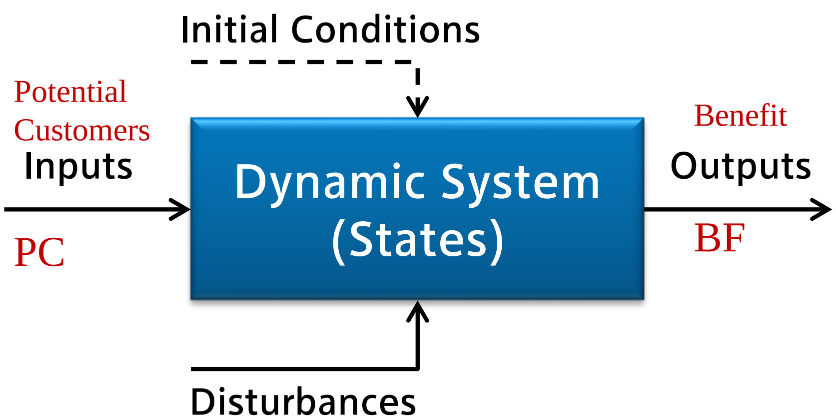
static formula

$$BF = Faktor \cdot PC$$

dynamic model

$$BF(t) = Function\left(PC(t), BF(t), t, Par\right)$$

Dynamical Systems



static formula

$$BF = Faktor \cdot PC$$

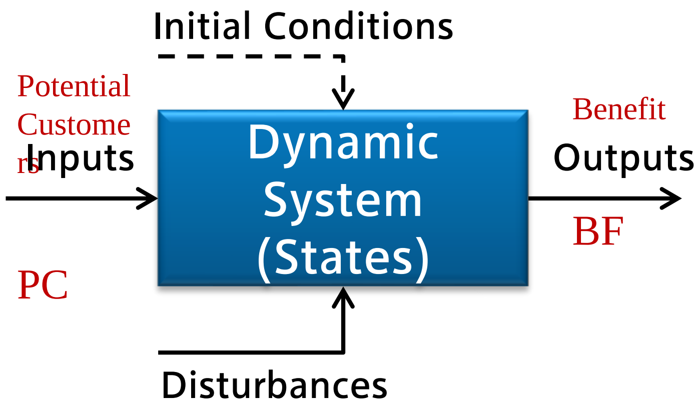
Calculation

dynamic model

$$BF(t) = Function(PC(t), BF(t), t, Par)$$

Simulation

Dynamical Systems

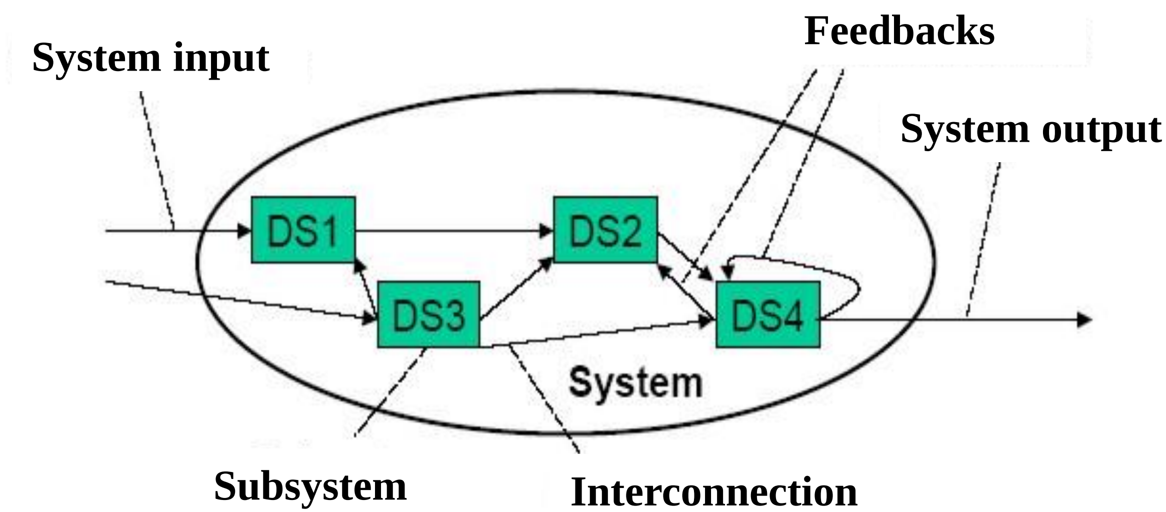


Dynamic mathematical model

$$BF(t) = Function(PC(t), BF(t), t, par)$$

Simulation

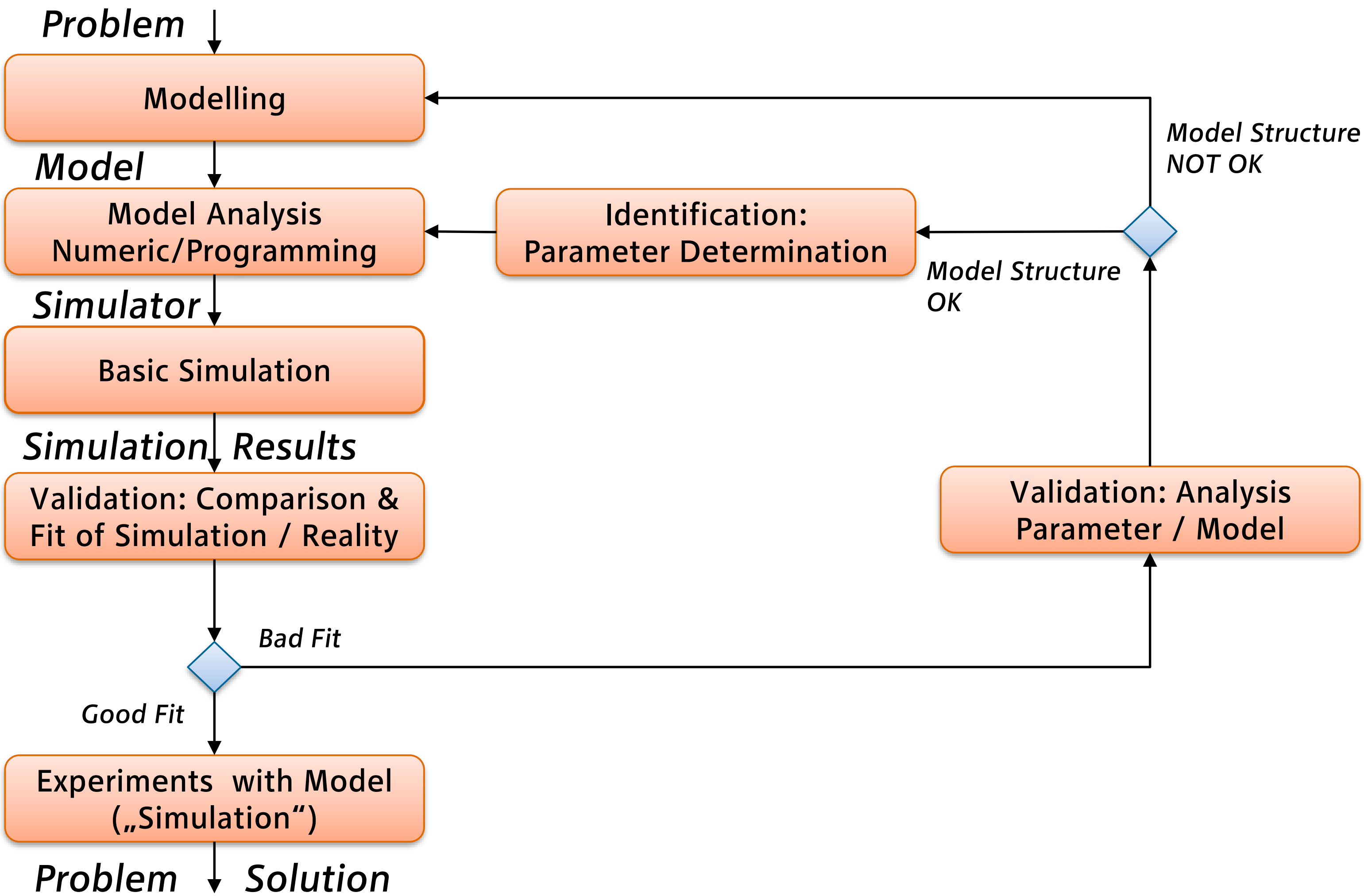
A dynamical system may consist of a set of components, which themselves are dynamical subsystems and which influence each other



not important

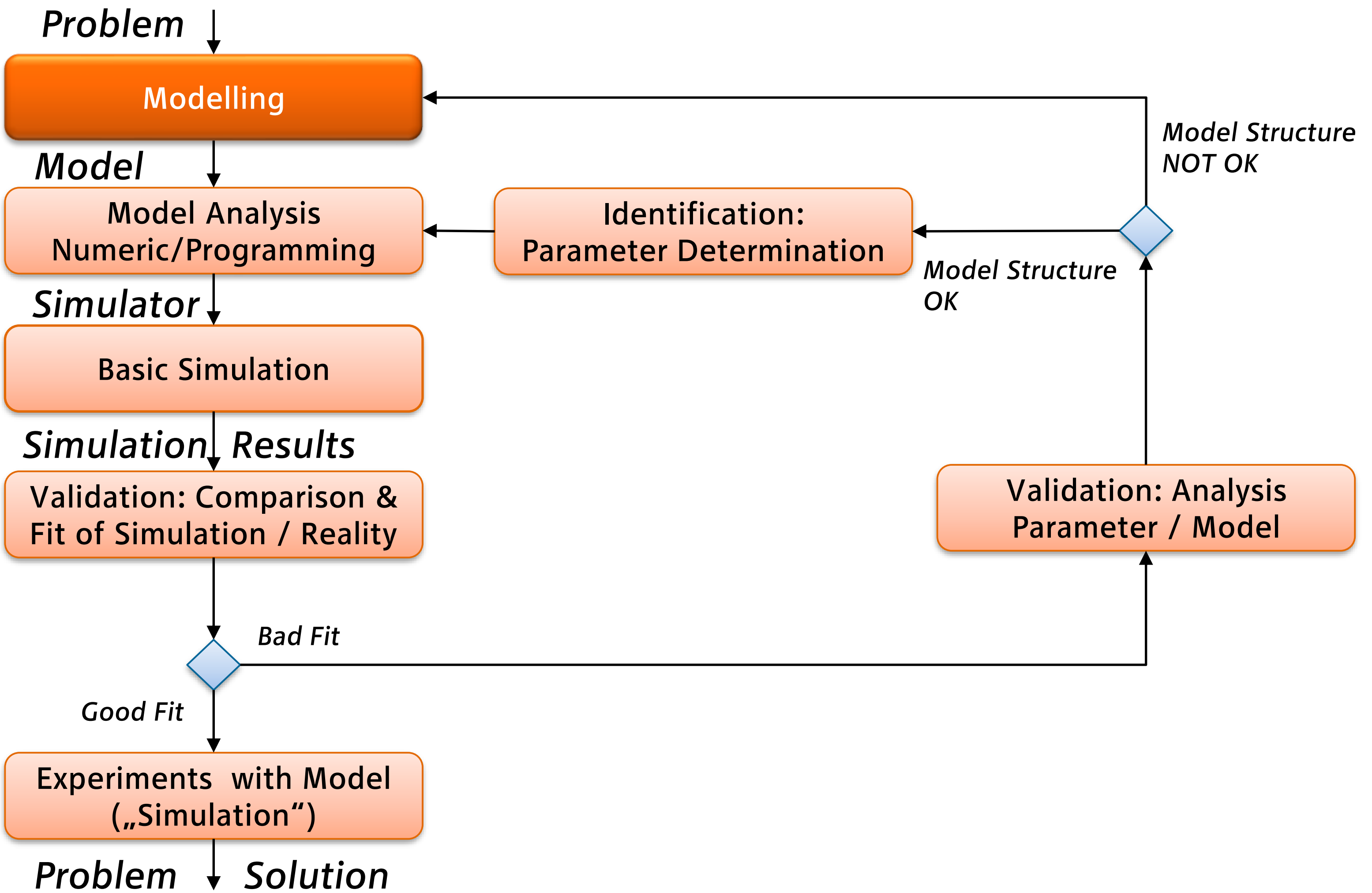
Simulation Circle

Mathematical Modelling and Simulation



Simulation Circle

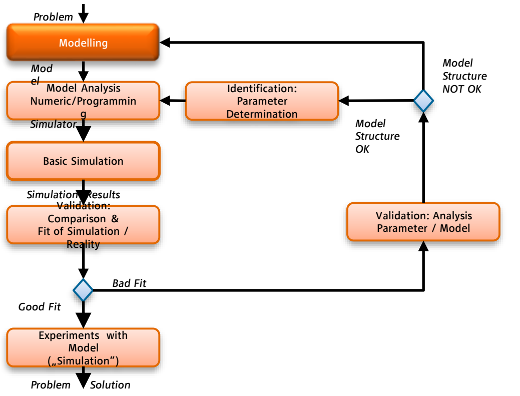
Mathematical Modelling and Simulation



What is a Model?

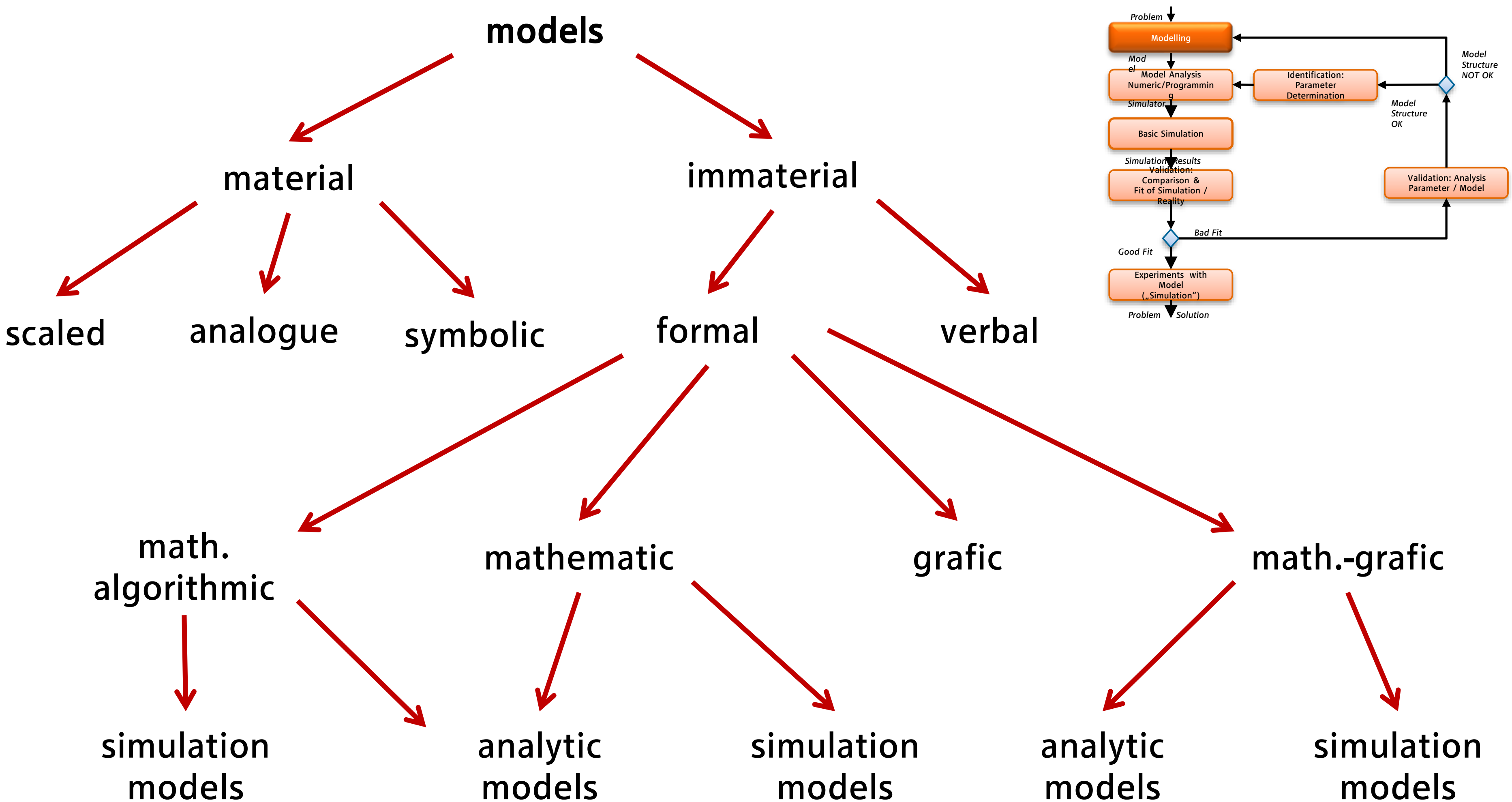
1. **Mapping** - A model is a representation of a natural or an artificial object.
2. **Reduction** - A model is usually simplified and does not have all attributes of the original object.
3. **Pragmatism** - A model is always created for a certain purpose, a certain subject and a certain time-span.

(Stachoviak 1973)

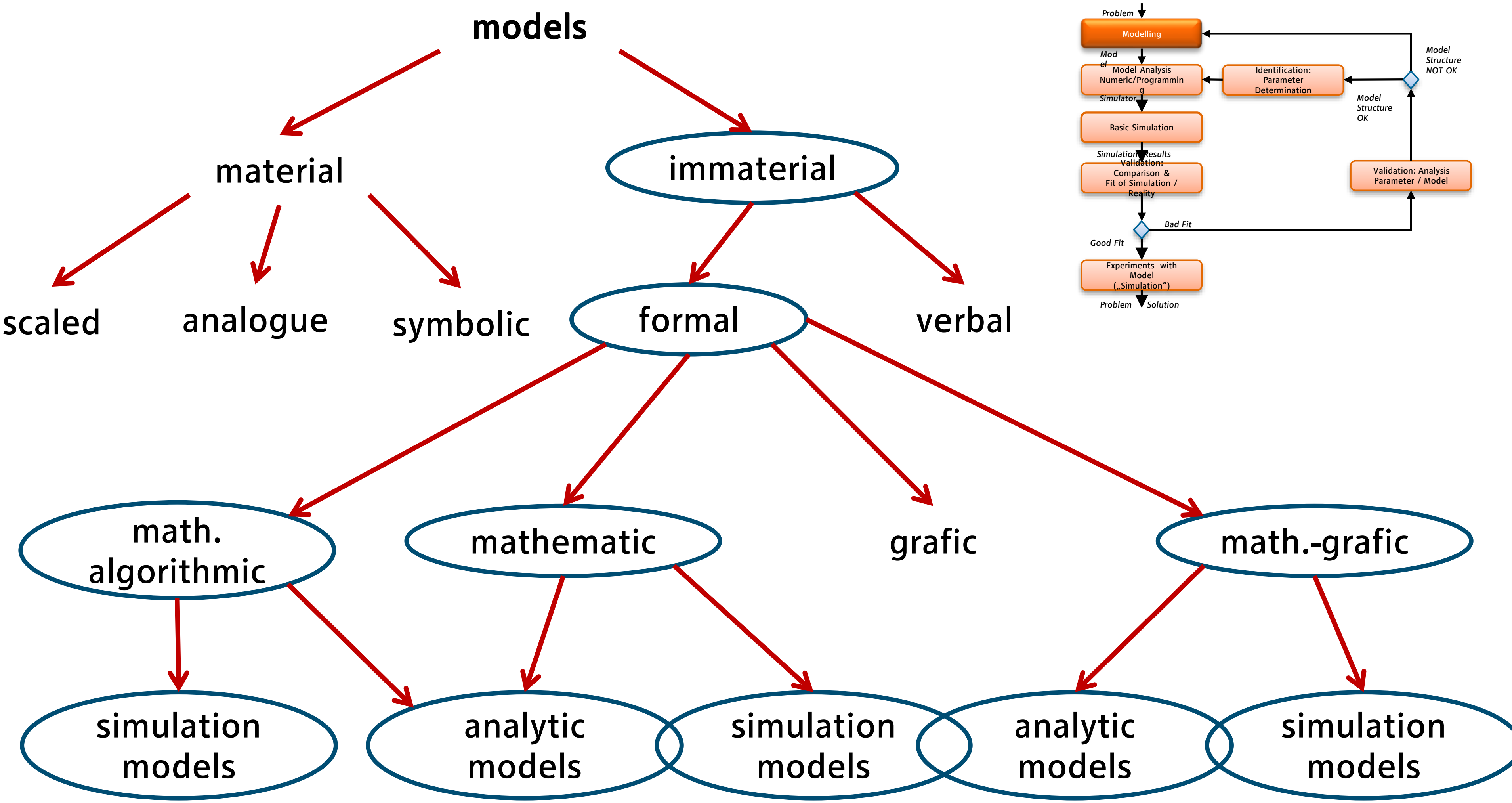


Model Classification

Mathematical Modelling and Simulation



Model Classification



Two Steps of Abstraction

- Structural Abstraction – Qualitative Knowledge
  Identification of system borders and states

- Phenomenological Abstraction – Quantitative Knowledge
  quantisation of states, identification of physical, economic, biologic, ... interactions in and with subsystems

Modelling vs Model

Mathematical Modelling and Simulation

Modelling Approach

- System Dynamics (SD)
- Transfer Functions (TF)
- Compartment Modelling
- Math. Formula
- Lagrange Formalism
- Port-based physical Modelling
- Difference Equation Modelling
- Cellular Automata Modelling
- Agent-based Modelling
- Event Graphs
- Process Flow

Model Type

- Ordinary Differential Equations (ODEs)
- Partial Differential Equations (PDEs)
- Differential Algebraic Equations (DAEs)
- Difference Equations (DEs)
- Cellular Automata (CAs)
- Agent-based Systems/Models (ABMs)
- Discrete Event Systems (DES)

Modelling vs Model

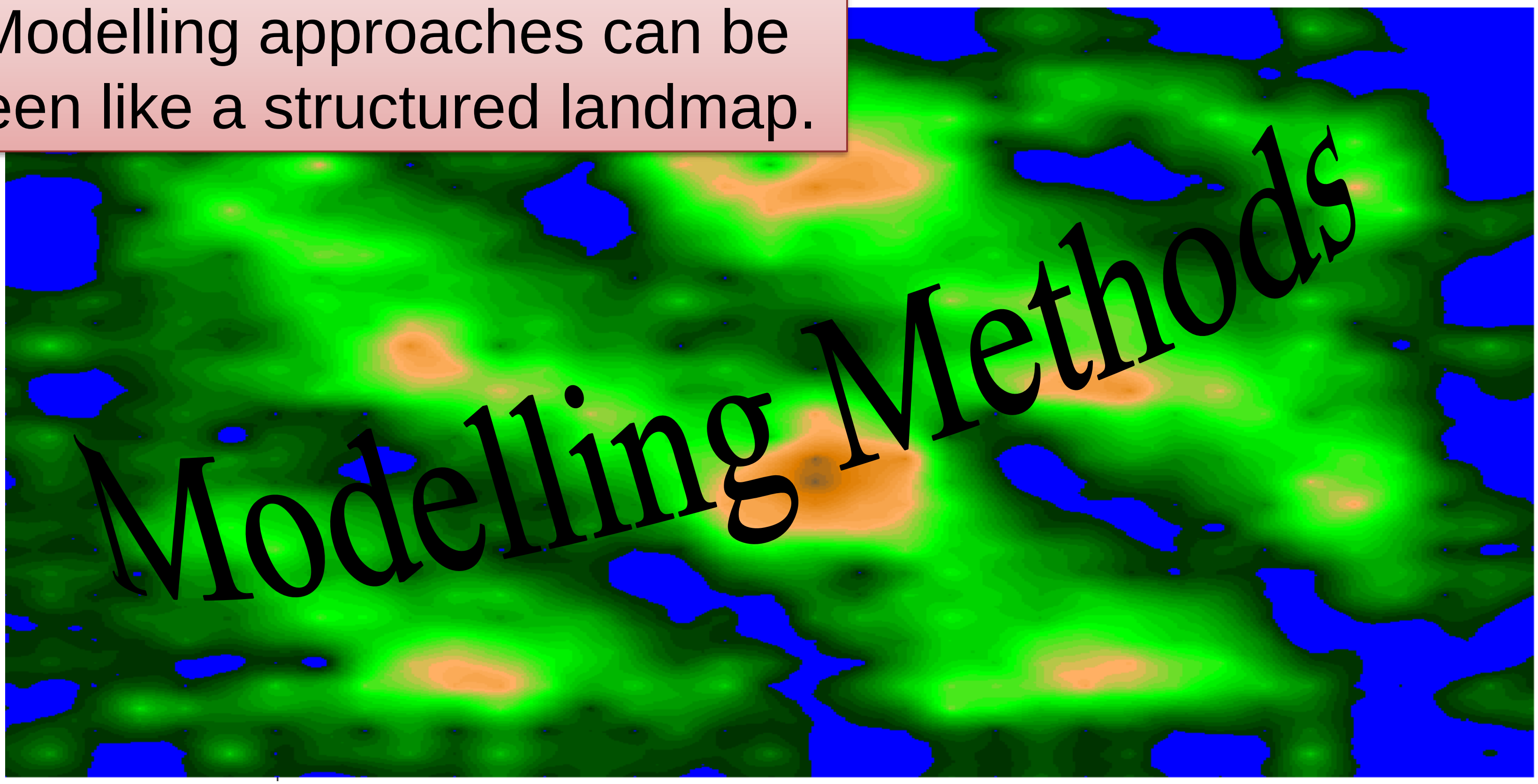
Mathematical Modelling and Simulation

Modelling Approach	Model Type
• System Dynamics (SD) • Transfer Functions (TF) • Compartment Modelling • Math. Formula • Lagrange Formalism • Port-based physical Modelling • Difference Equation Modelling • Cellular Automata Modelling • Agent-based Modelling • Event Graphs • Process Flow	• Ordinary Differential Equations (ODEs) • Partial Differential Equations (PDEs) • Differential Algebraic Equations (DAEs) • Difference Equations (DEs) • Cellular Automata (CAs) • Agent-based Systems/Models (ABMs) • Discrete Event Systems (DES)

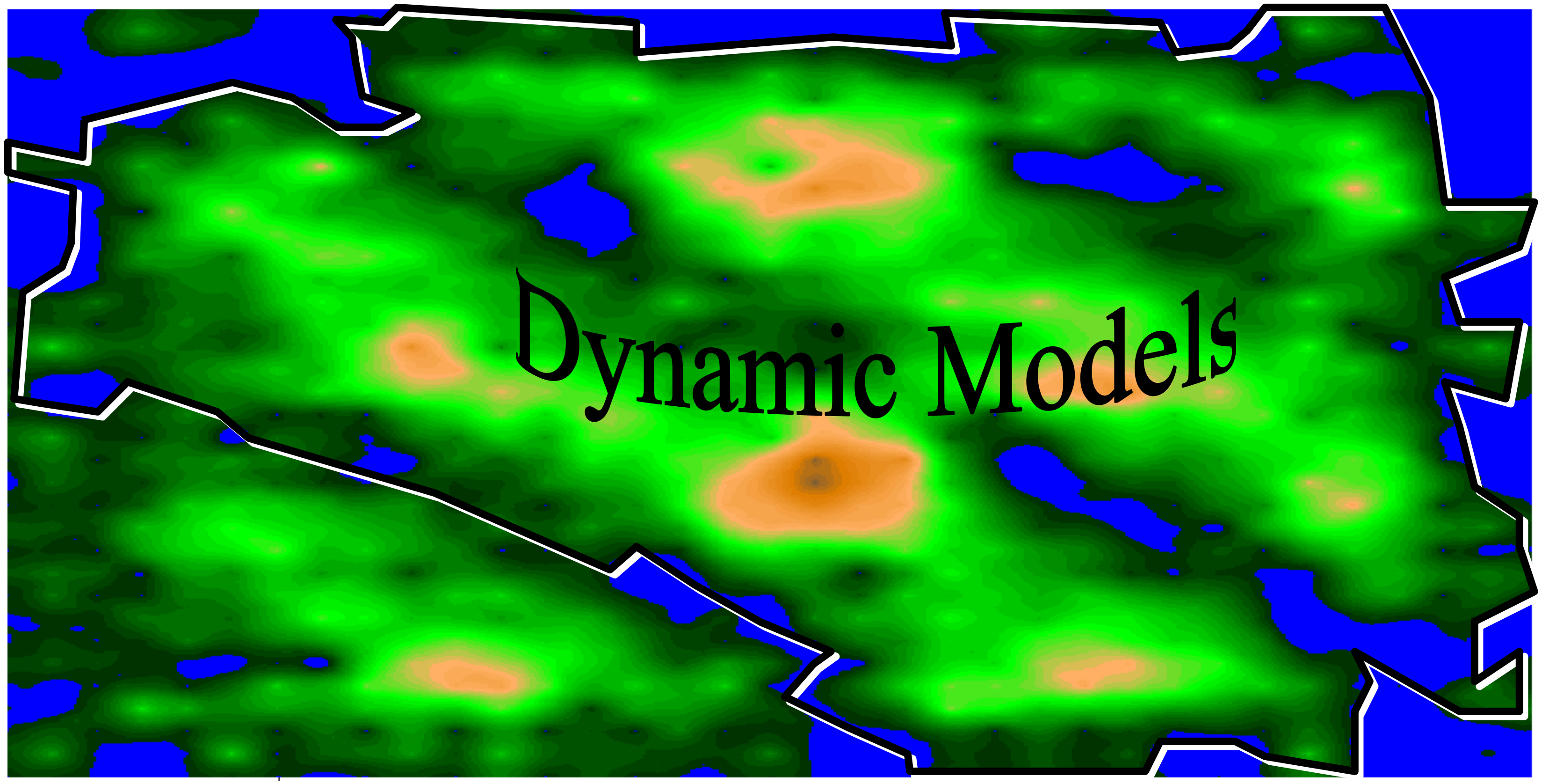
Landmap of Modelling Methods

Mathematical Modelling and Simulation

The variety of different Modelling approaches can be seen like a structured landmap.



Landmap of Modelling Methods – Dynamic Models



Dynamic Models – Time Discrete/Continuous

Mathematical Modelling and Simulation

Neglecting quantum-mechanics (space as well as) time can be seen to be a continuous number.

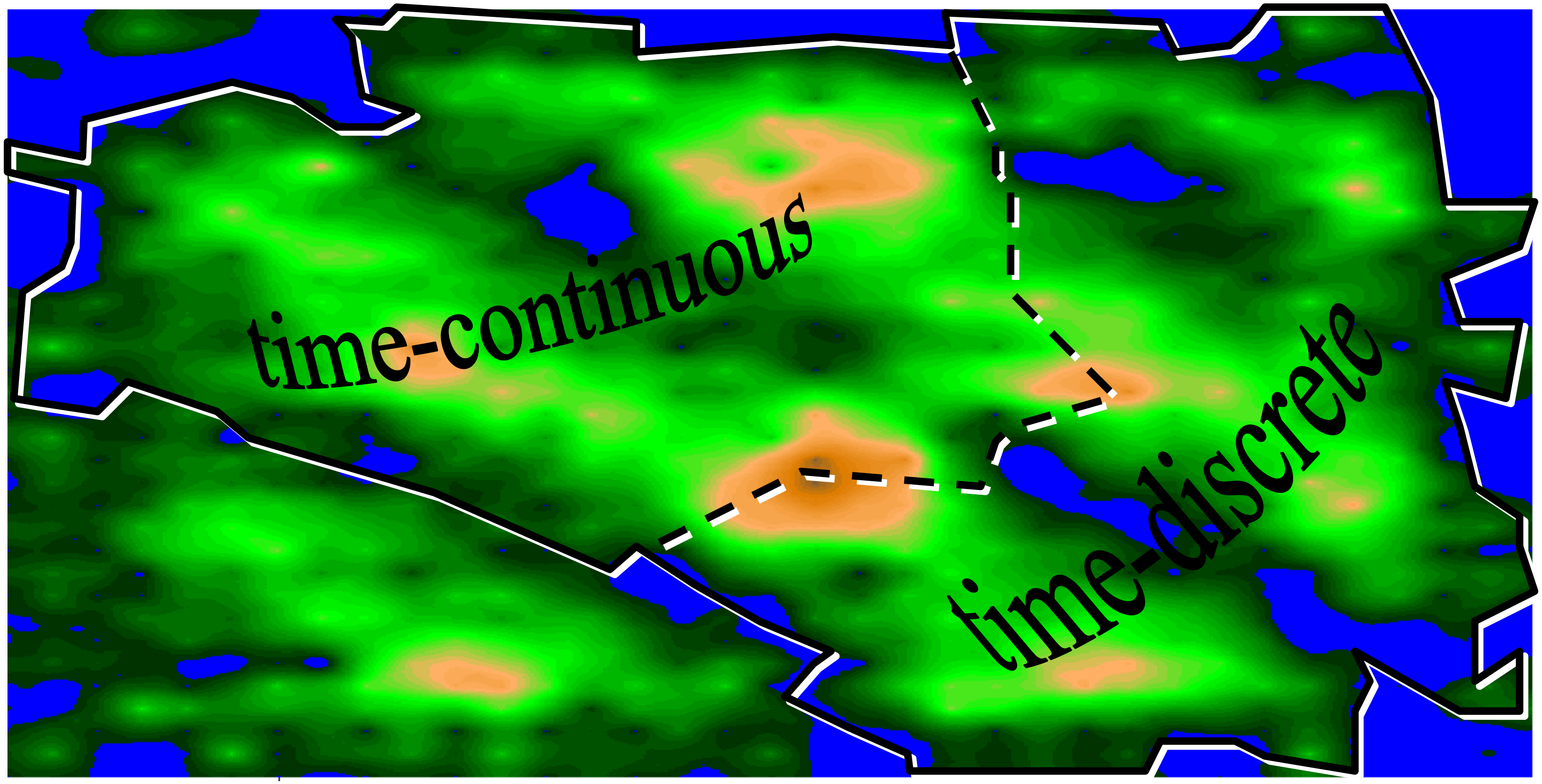
- A model is called time-continuous if the output value of the model can be calculated at any time ($\approx t \in \mathbb{R}$).
- In the opposite a model is called time-discrete if values are only calculated at a finite number of predefined timesteps ($\approx t \in \mathbb{N}$).

Dynamic Models – Time Discrete/Continuous

- Usually time-continuous models are preferred to time-discrete models, but the simulation process is usually more difficult.
- Yet, there are processes in real world for which time continous models are not necessary or even dont make sense.
- Very often, time-continuous models cannot be simulated continously. So they need to be reformalised in a time-discrete manner – this process is called discretisation.

Landmap of Modelling Methods – Time Discrete / Continous

Mathematical Modelling and Simulation



Dynamic Models – Value Discrete/Continuous

Similar to time-discrete/continuous, also output values can be determined discrete or continuously.

- **Value-discrete**:
  - Number of passengers on a plane
  - Number of cars searching for a parking spot.
- **Value-continuous**:
  - Voltage/Current in an Electrical Circuit
  - Angular Velocity of a Pendulum

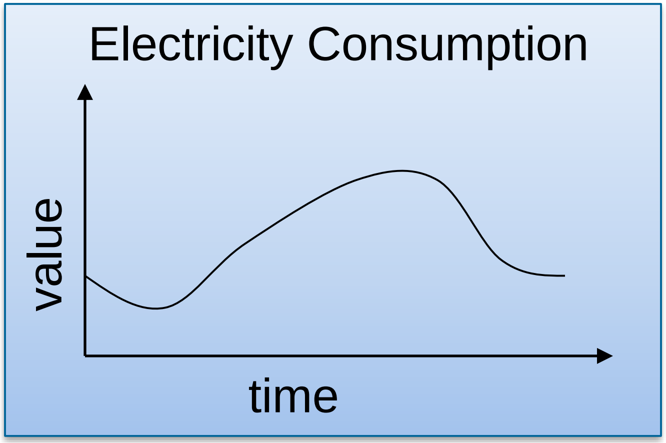
Dynamic Models – Value Discrete/Continuous

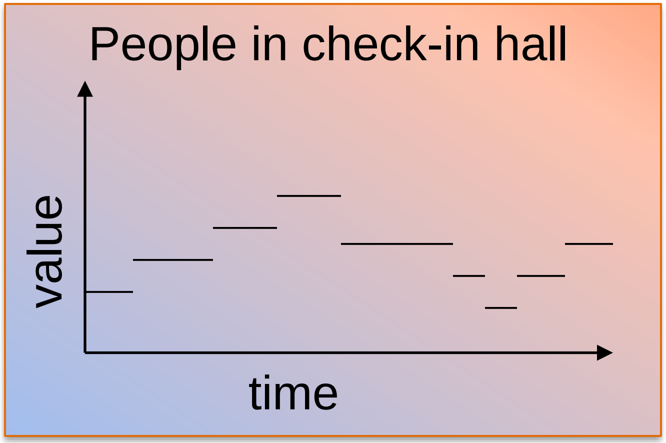
- Although simulation output is expected to be continuous/discrete, it is not necessarily modelled in a continuous/discrete way.

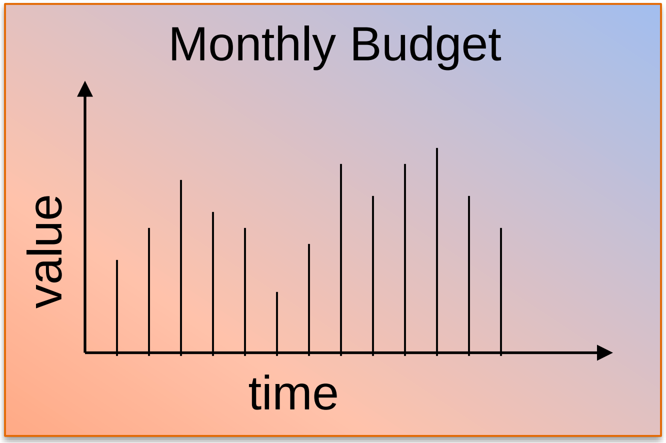
  E.g.:
  Population of a country is a discrete number…
  … yet it can be modelled by a continous model

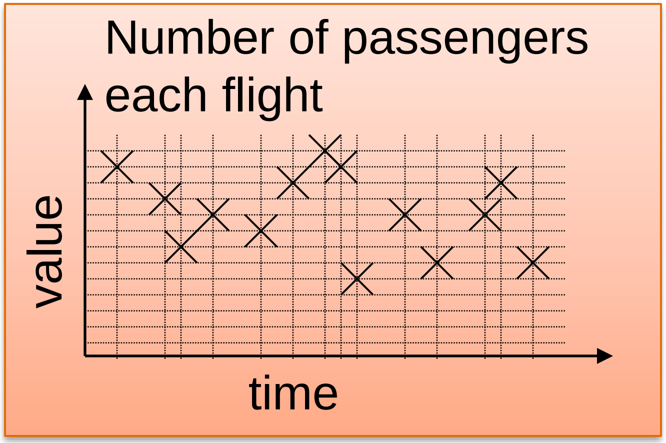
  It requires a correct result interpretation!

Examples –Discrete/Continuous



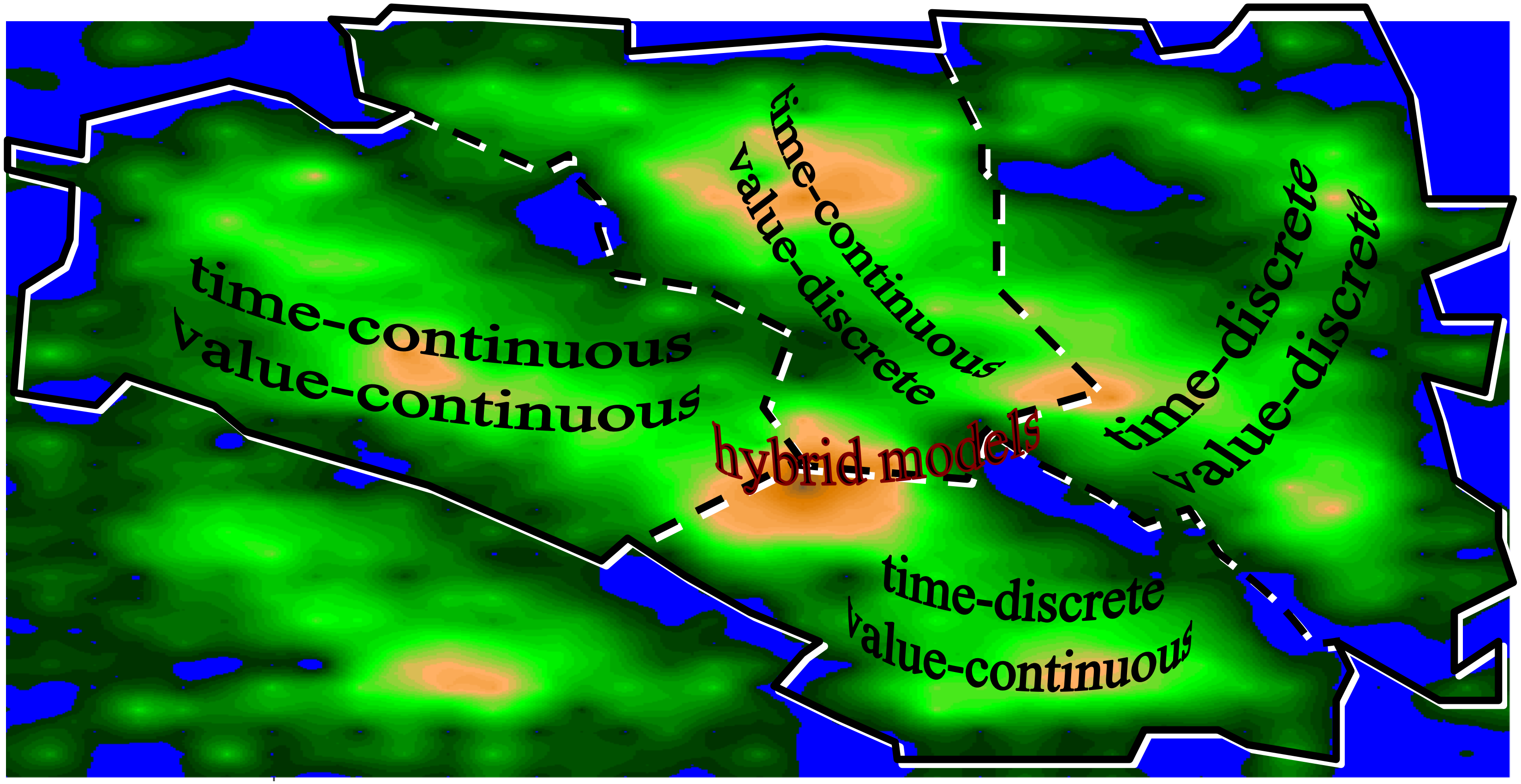






Landmap of Modelling Methods – Discrete / Continous

Mathematical Modelling and Simulation



Model Procedures

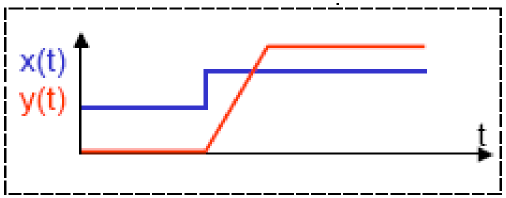
Mathematical Modelling and Simulation

System

Theoretical Modelling	Experimental Modelling
Deductive Analysis	Inductive Analysis
Modelling by Laws and Rules	Modelling by using models with observed behaviour
White Box Modelling	Black Box Modelling

Mathematical Model

Ohms Law
Newton‘s Law
Supply/Demand Law

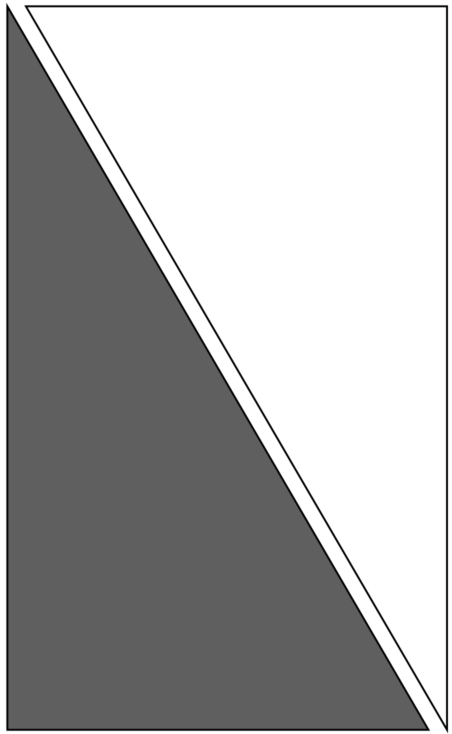


Application vs. Modelling Approach

Mathematical Modelling and Simulation

White Box Modeling

- Electrotechnique
- Mechanics
- Environment
- Medicine
- Economy
- Sociology



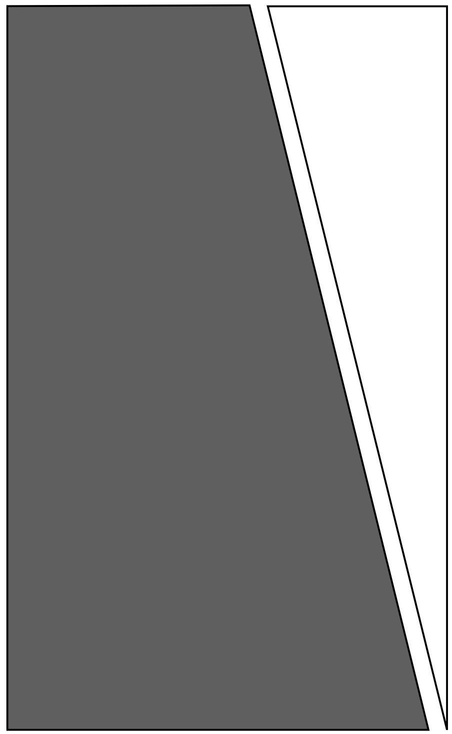
- Laws
- Laws and Observations
- Laws and Observations
- Observations and Characterisation
- Observations and Characterisation

Black Box Modeling

Mathematical Modelling and Simulation

From Deduction to Induction

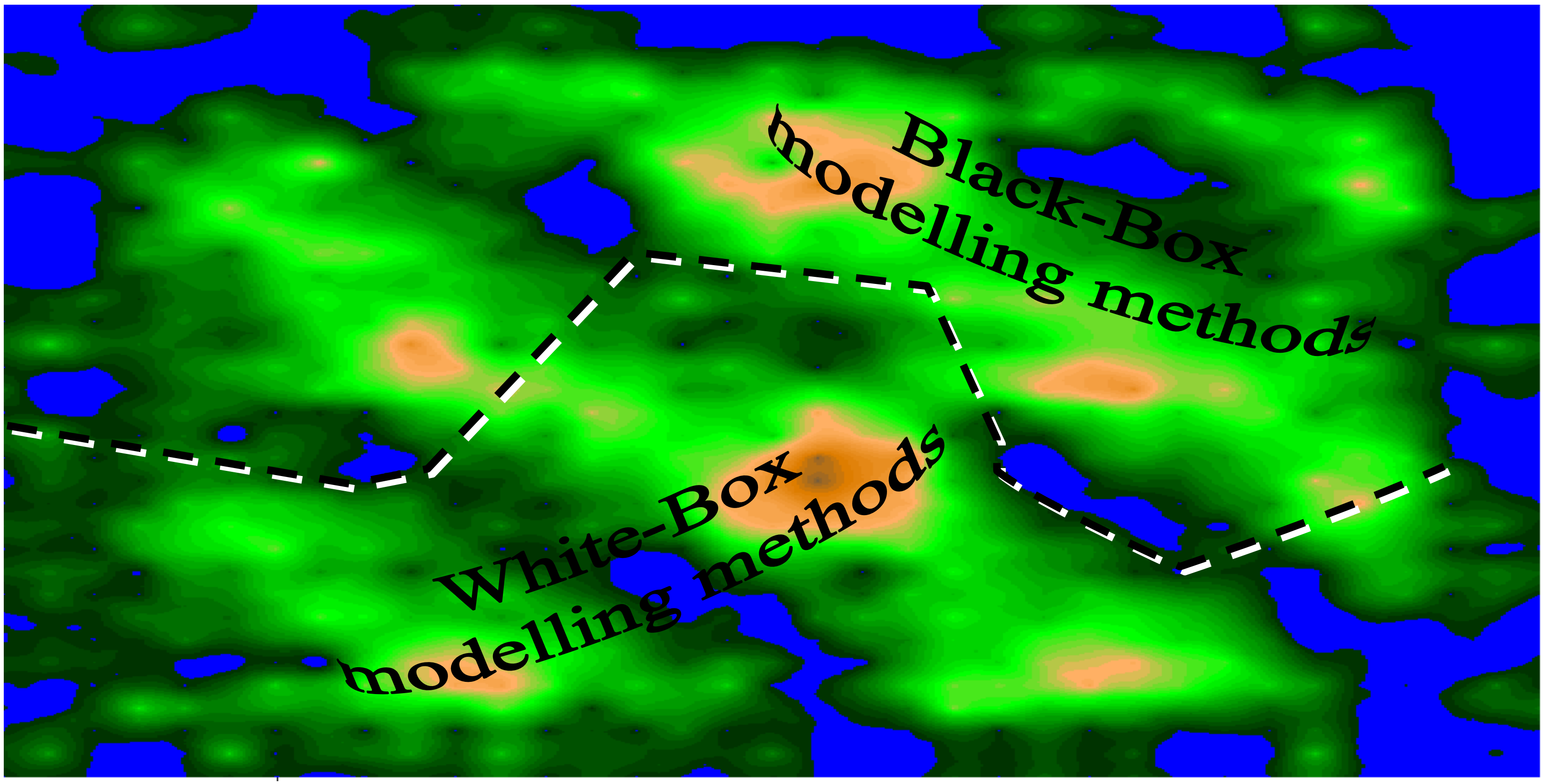
- Electrotechnique
- Mechanics
- Environment
- Medicine
- Economy
- Sociology



- Laws
- Laws and Observations
- Laws and Observations
- Observations and Characterisation
- Observations and Characterisation

Deductive models may contain too many parameters – problems with identification

Landmap of Modelling Methods – Discrete / Continous



Stochastic/Deterministic

- If the output of the simulation of a model is uniquely defined by input parameters, initial conditions and model parameters the model is called **deterministic**.
- Otherwise it is called **stochastic**.

Results:
14.2
14.2
14.2
14.2
...

$\Rightarrow$ deterministic

Results:
14.3
14.6
13.9
14.2
...

$\Rightarrow$ stochastic

Stochastic/Deterministic

Mathematical Modelling and Simulation

Stochastic models are necessary...

- ... if random effects are included in the system.
  → coin toss, rolling a dice, ...

- ... if emelents of the system are too complex to be described by deterministic rules.

→ human behaviour, problems at system borders,...

Stochastic/Deterministic

Stochastic models are necessary...

- ... if random effects are included in the system.
  → coin toss, rolling a dice, ...

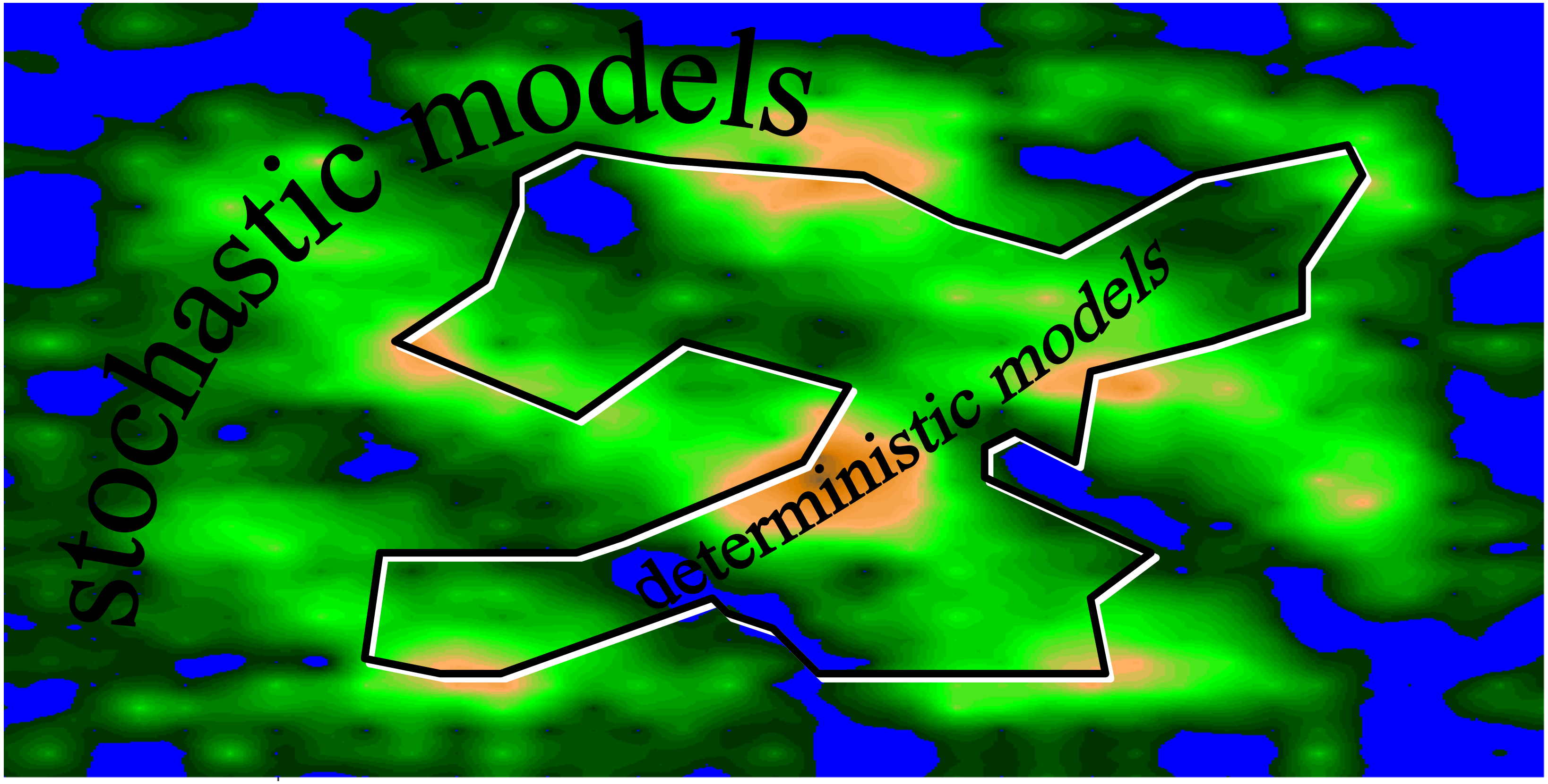
- ... if emelents of the system are too complex to be described by deterministic rules.

→ human behaviour, problems at system borders,...

→ coin toss, rolling a dice, ...

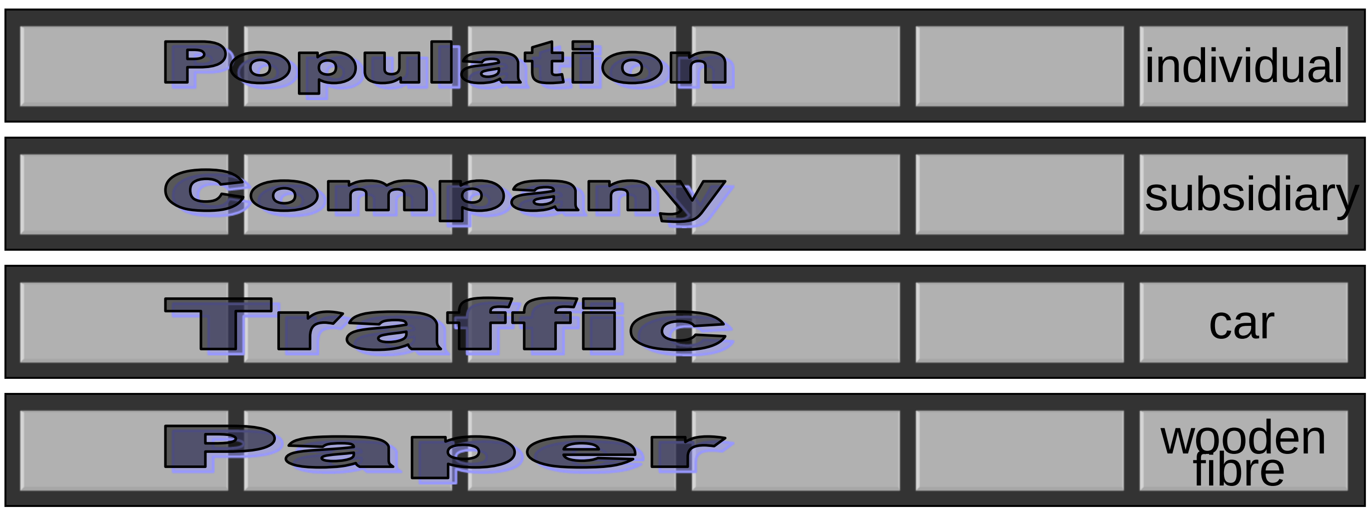
Landmap of Modelling Methods – Discrete / Continous

Mathematical Modelling and Simulation



Microscopic/Macroscopic Models

- If systems consist of a big set of similar subsystems...

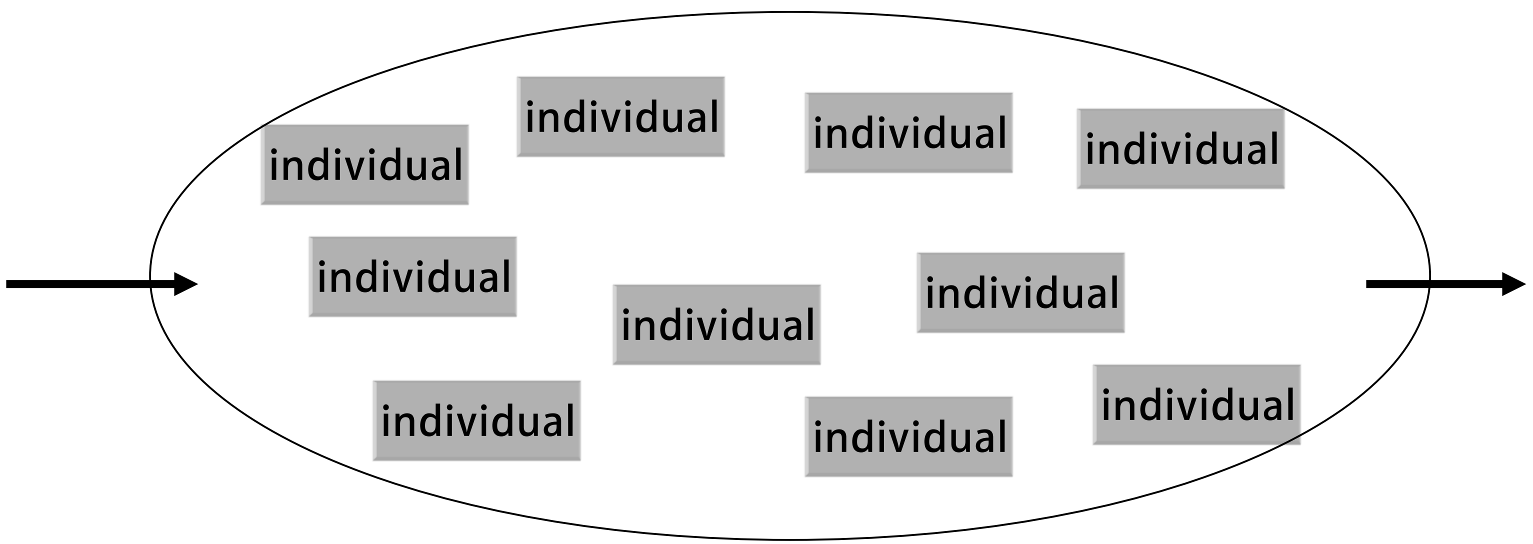


... the question arises whether a micro- meso- or macroscopic model should be used.

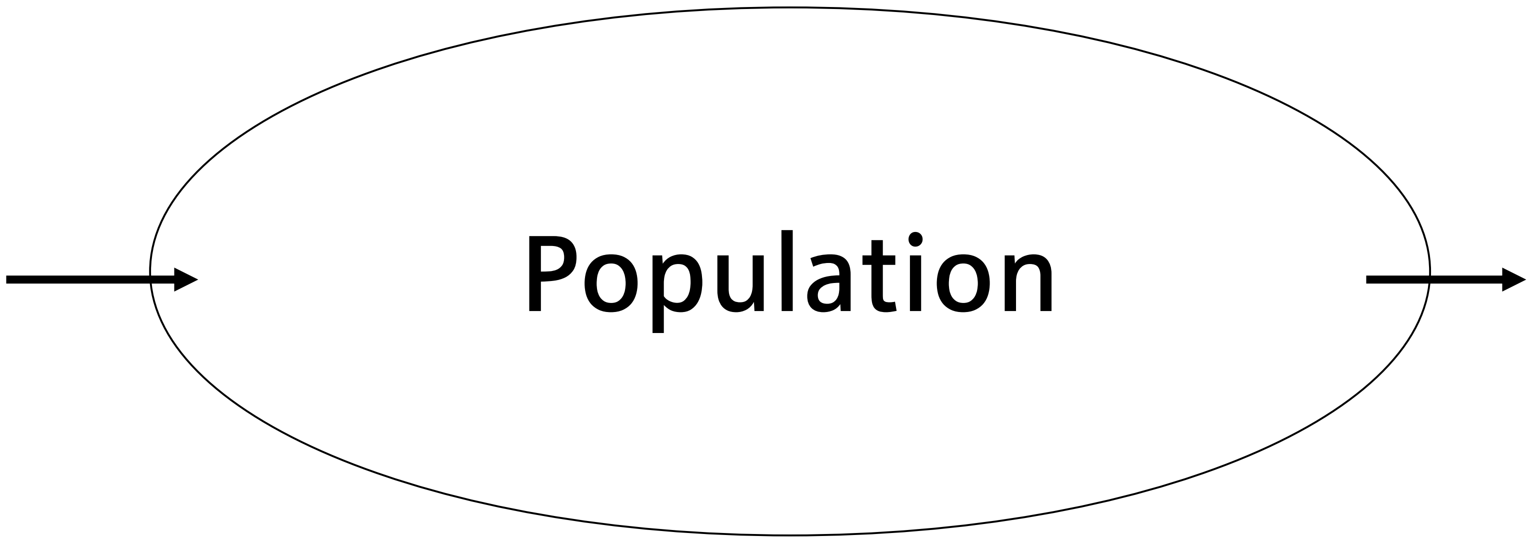
Microscopic/Macroscopic Models

Mathematical Modelling and Simulation

- Microscopic models treat each subsystem as an individual model. Finally they are linked in order to model the whole system.

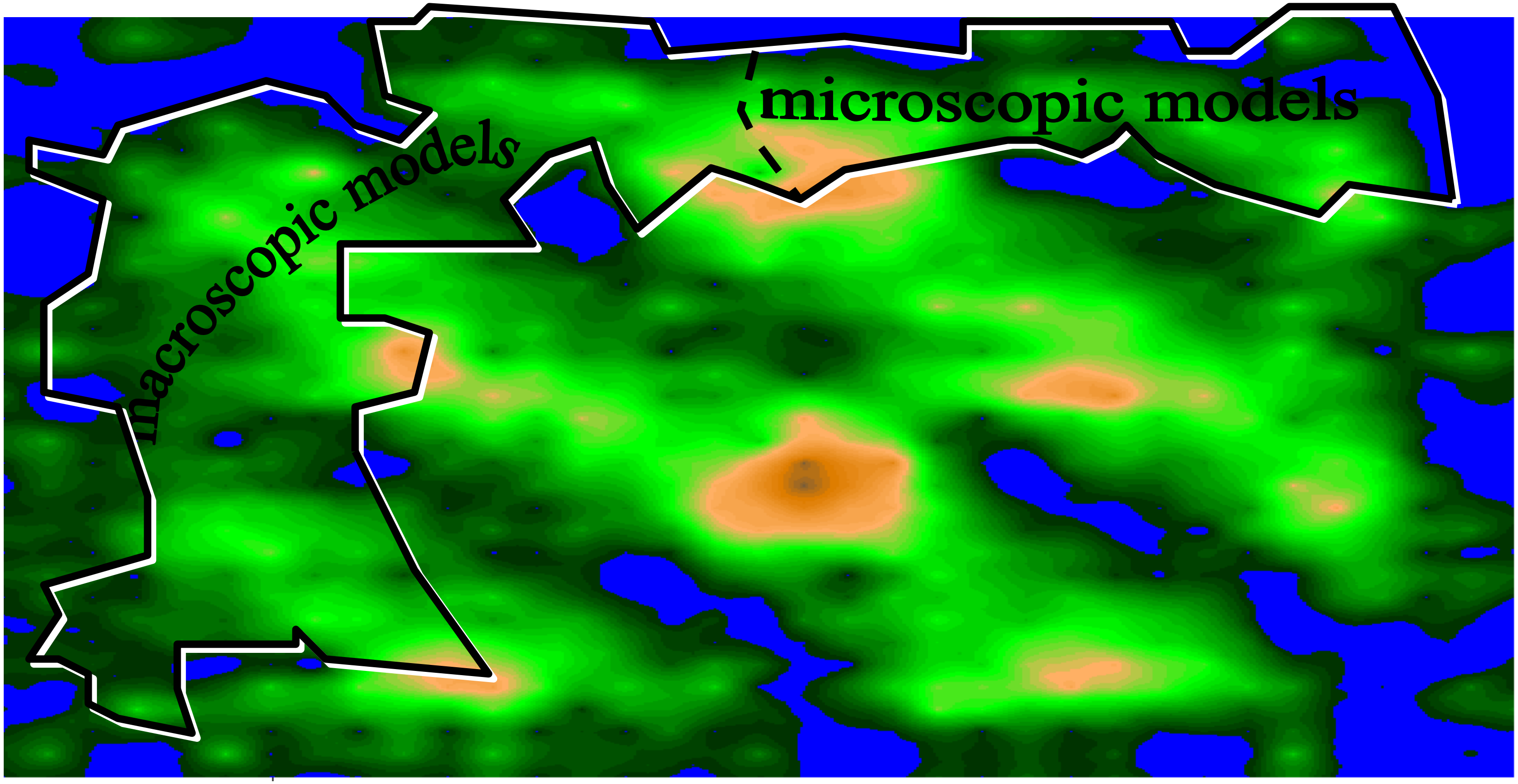


- Macroscopic models treat the whole system, neglecting the fact, that it consists of subsystems.



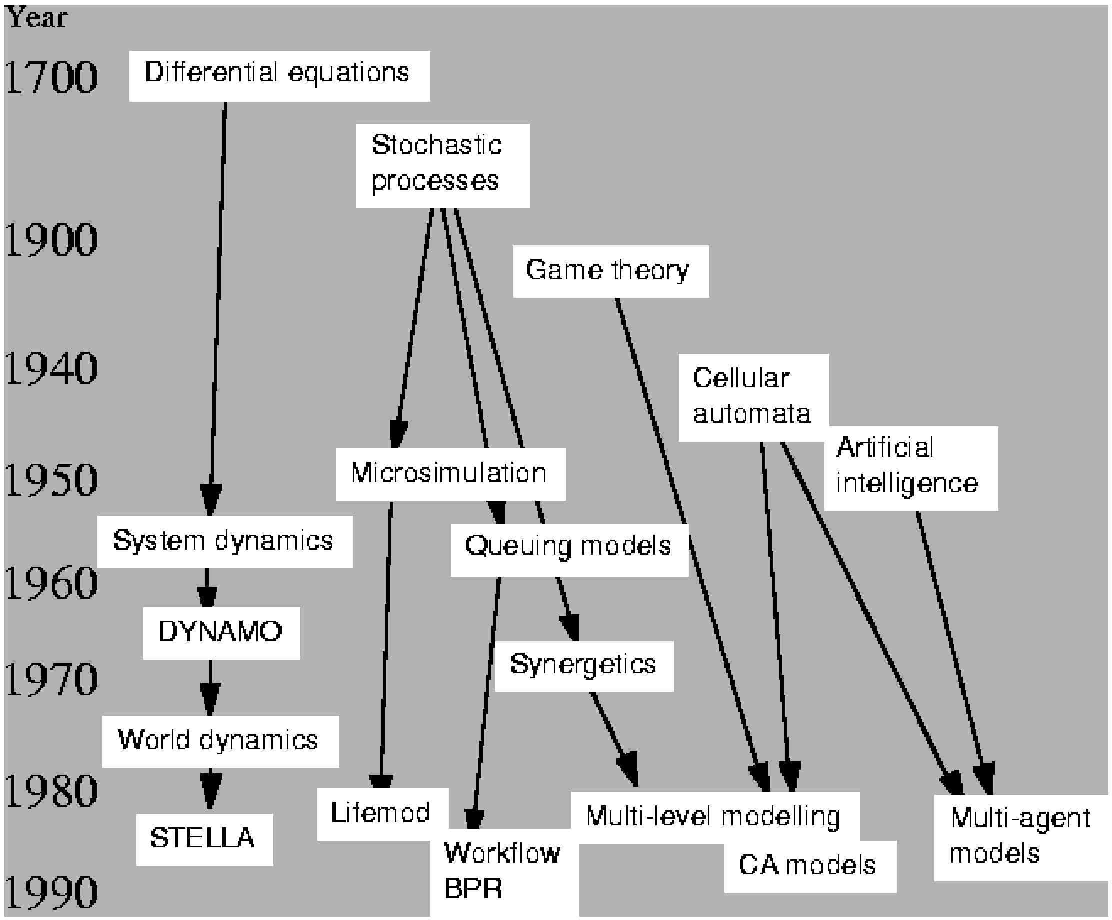
Landmap of Modelling Methods – Microscopic/Macroscopic

Mathematical Modelling and Simulation



Approaches for Soft Sciences Simulation

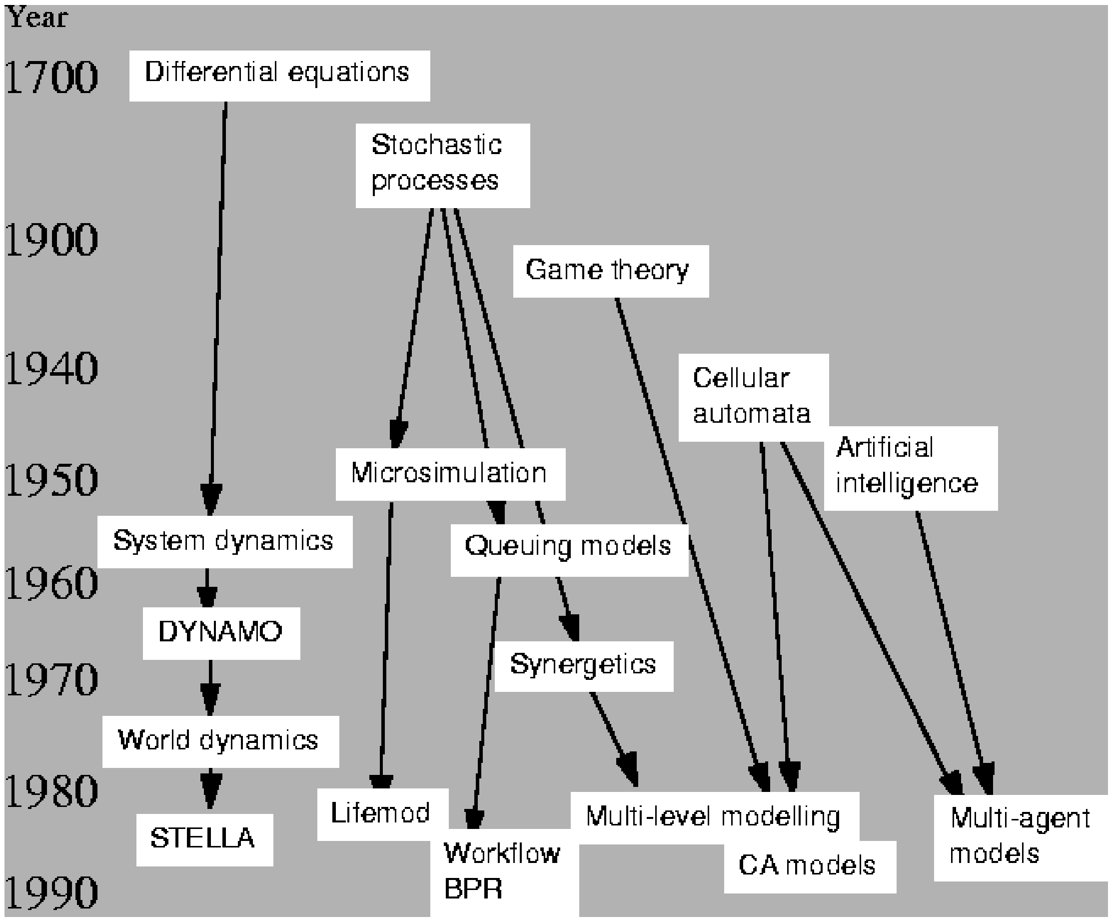
Mathematical Modelling and Simulation



(Troitzsch)

Approaches for Soft Sciences Simulation

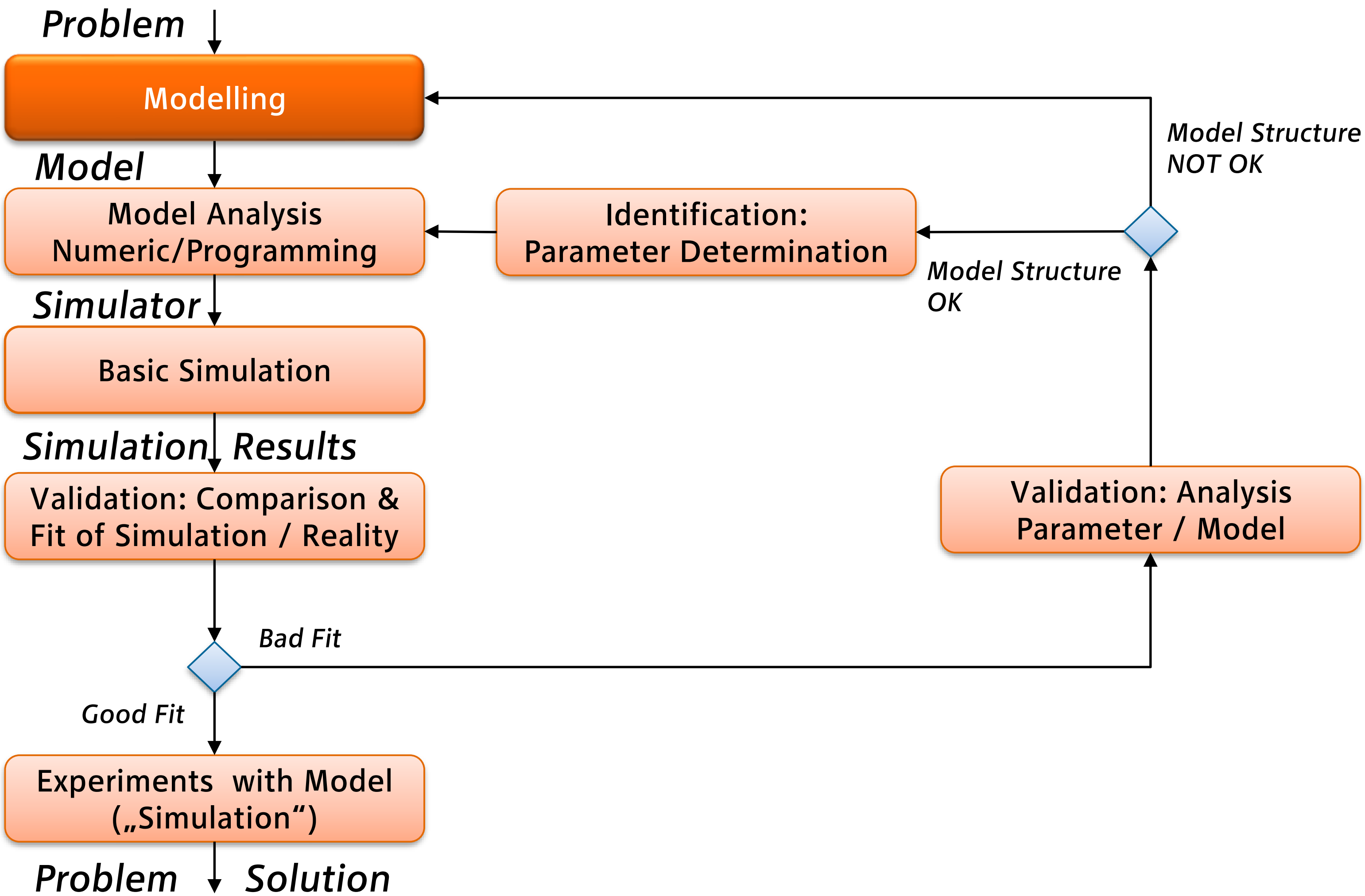
Mathematical Modelling and Simulation



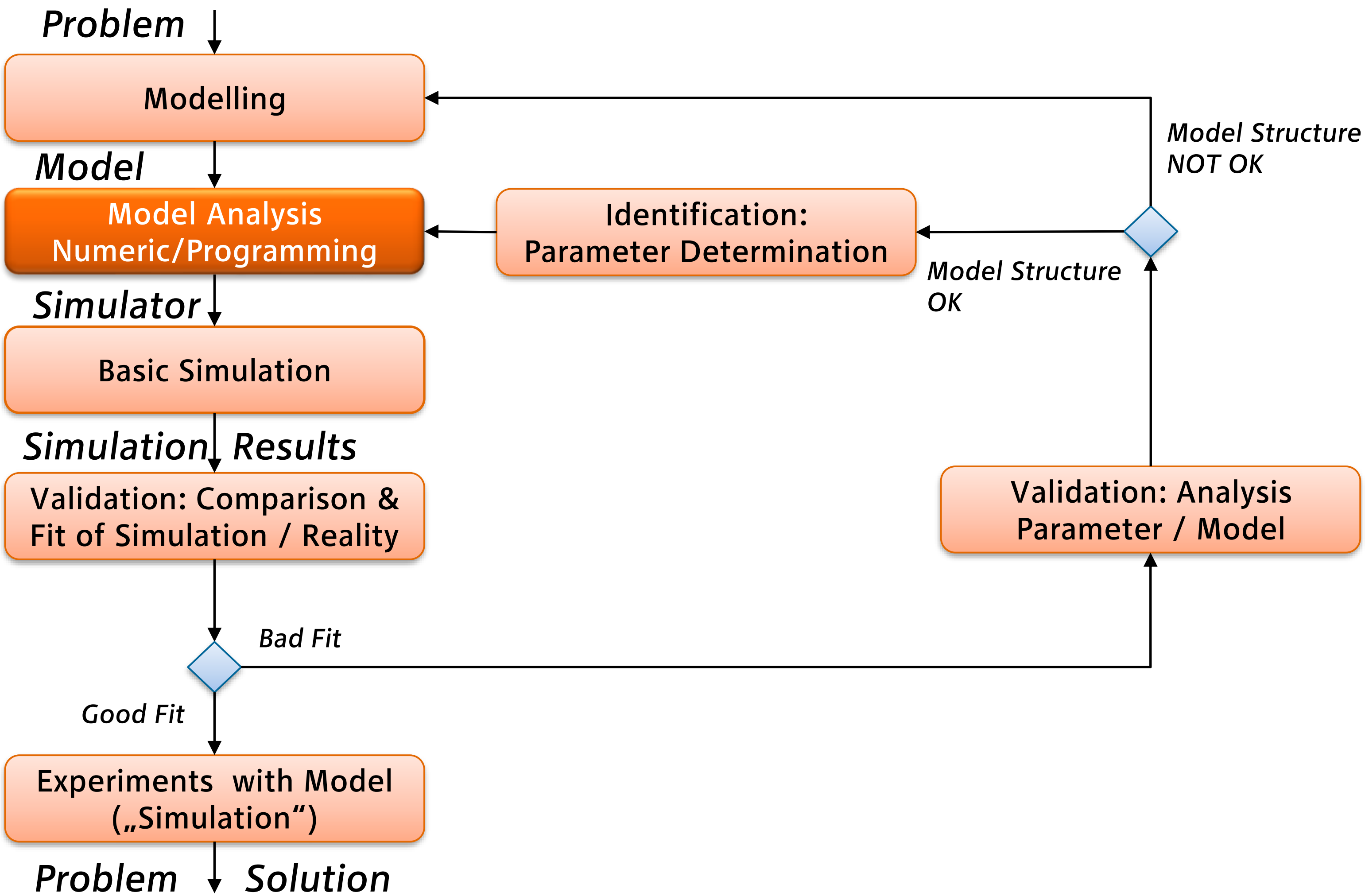
Aggregated ⟷ Distributed (Troitzsch)

Simulation Circle

Mathematical Modelling and Simulation

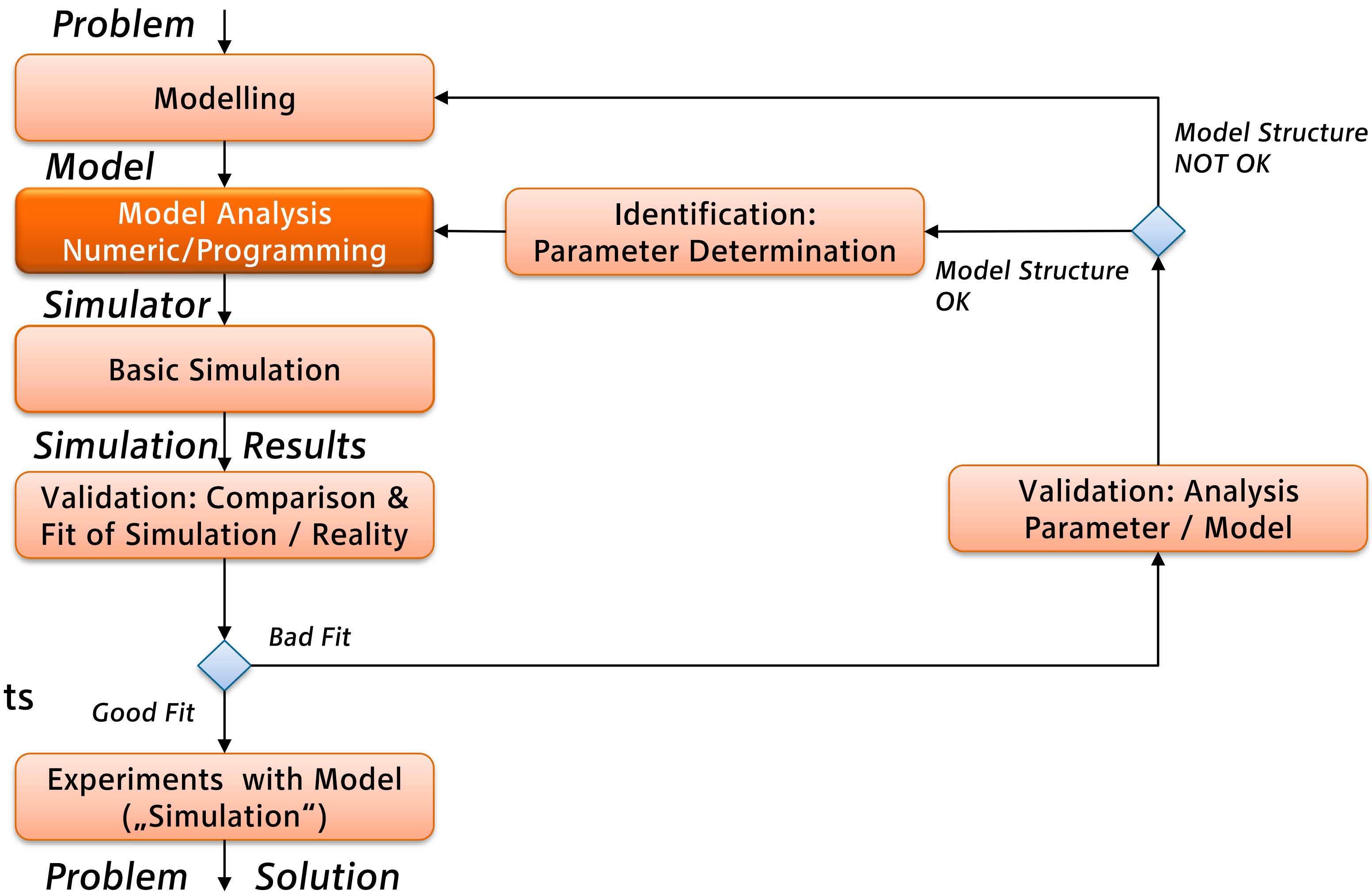


Simulation Circle



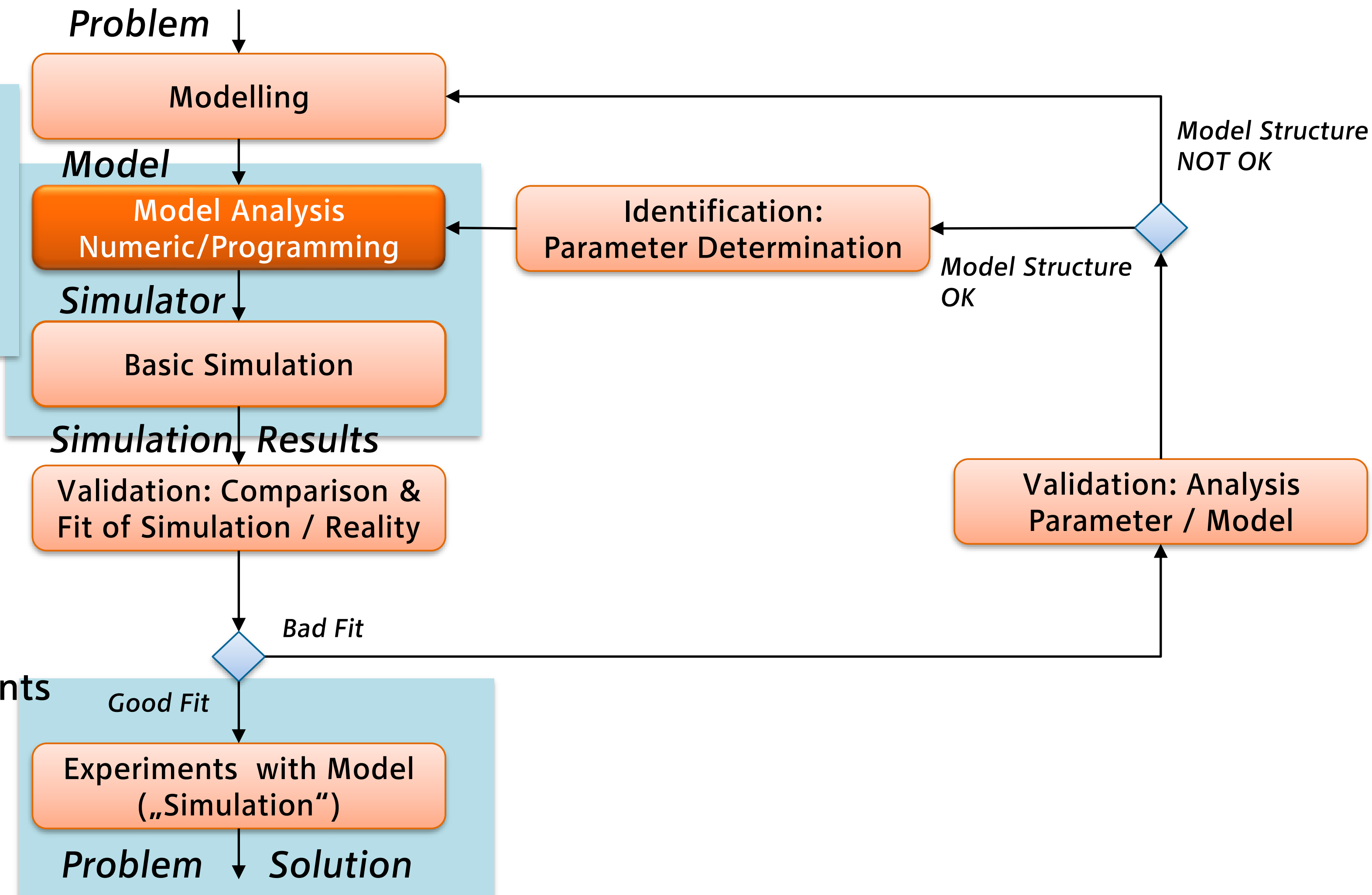
Simulation Circle

- simulation software
- simulation languages
- Simulators
- Simulation systems
- Simulation Environments



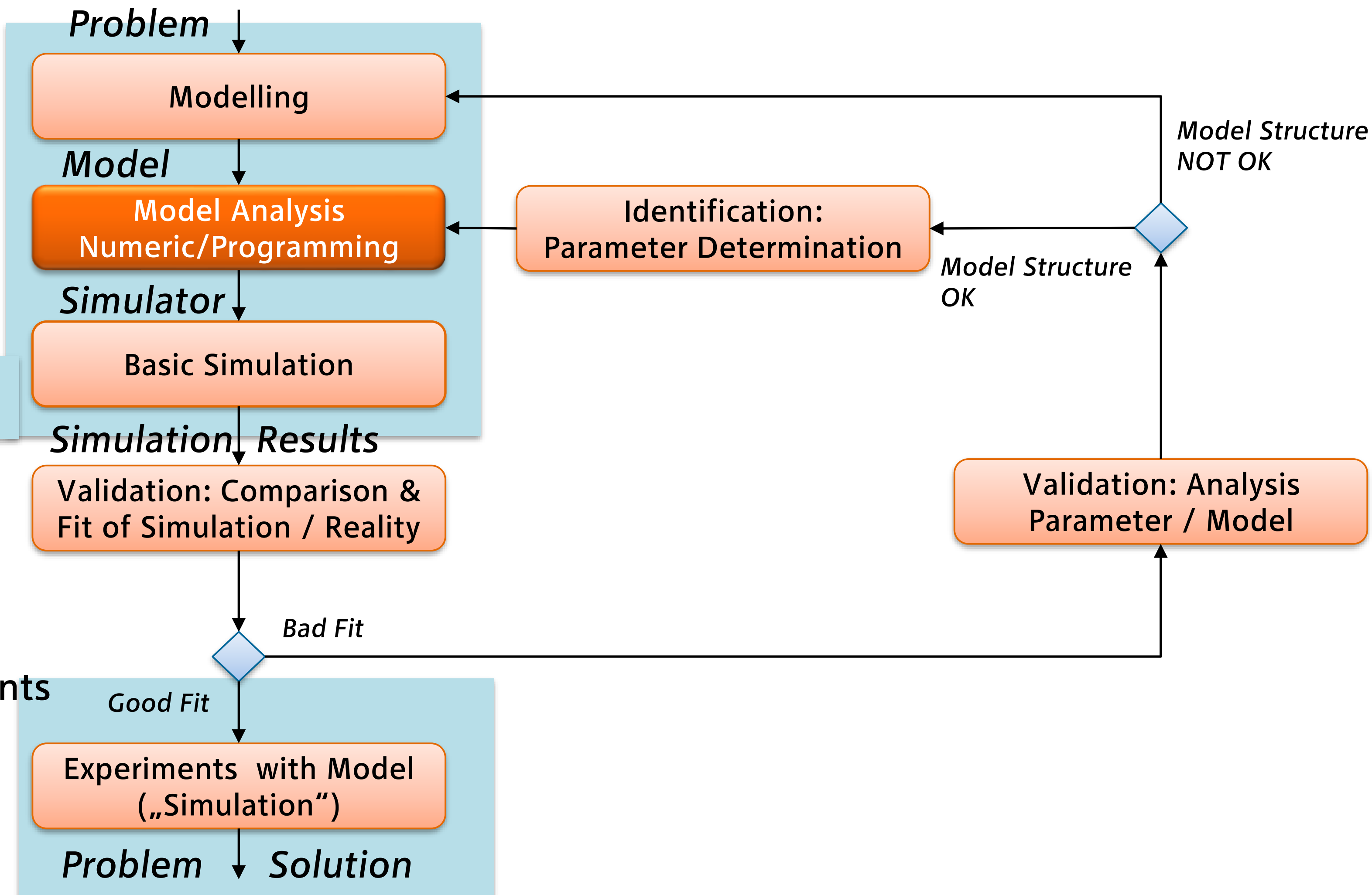
Simulation Circle

- simulation software
- simulation languages
- Simulators
- Simulation systems
- Simulation Environments



Simulation Circle

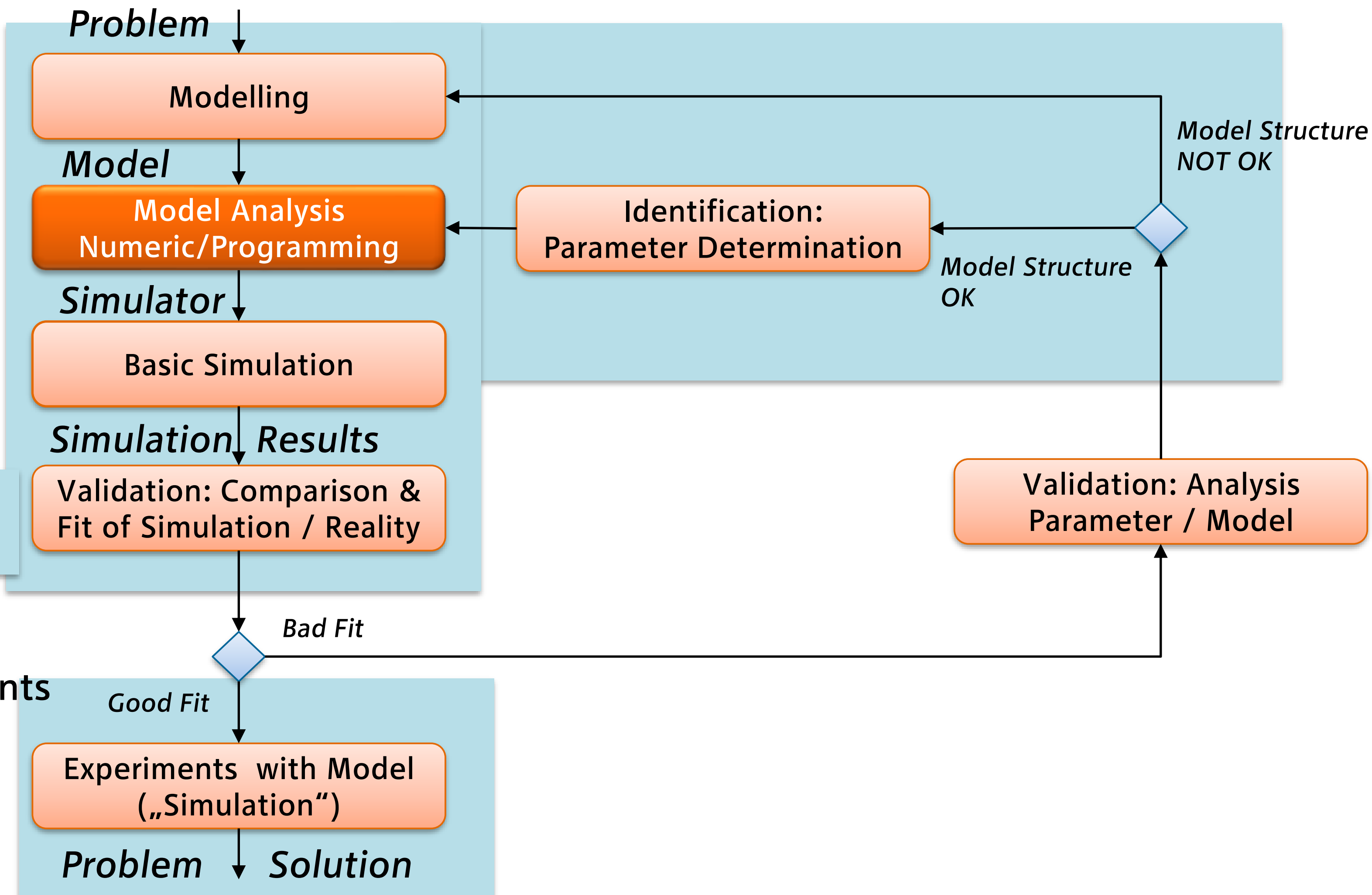
- simulation software
- simulation languages
- **Simulators**
- Simulation systems
- Simulation Environments



Simulation Circle

Mathematical Modelling and Simulation

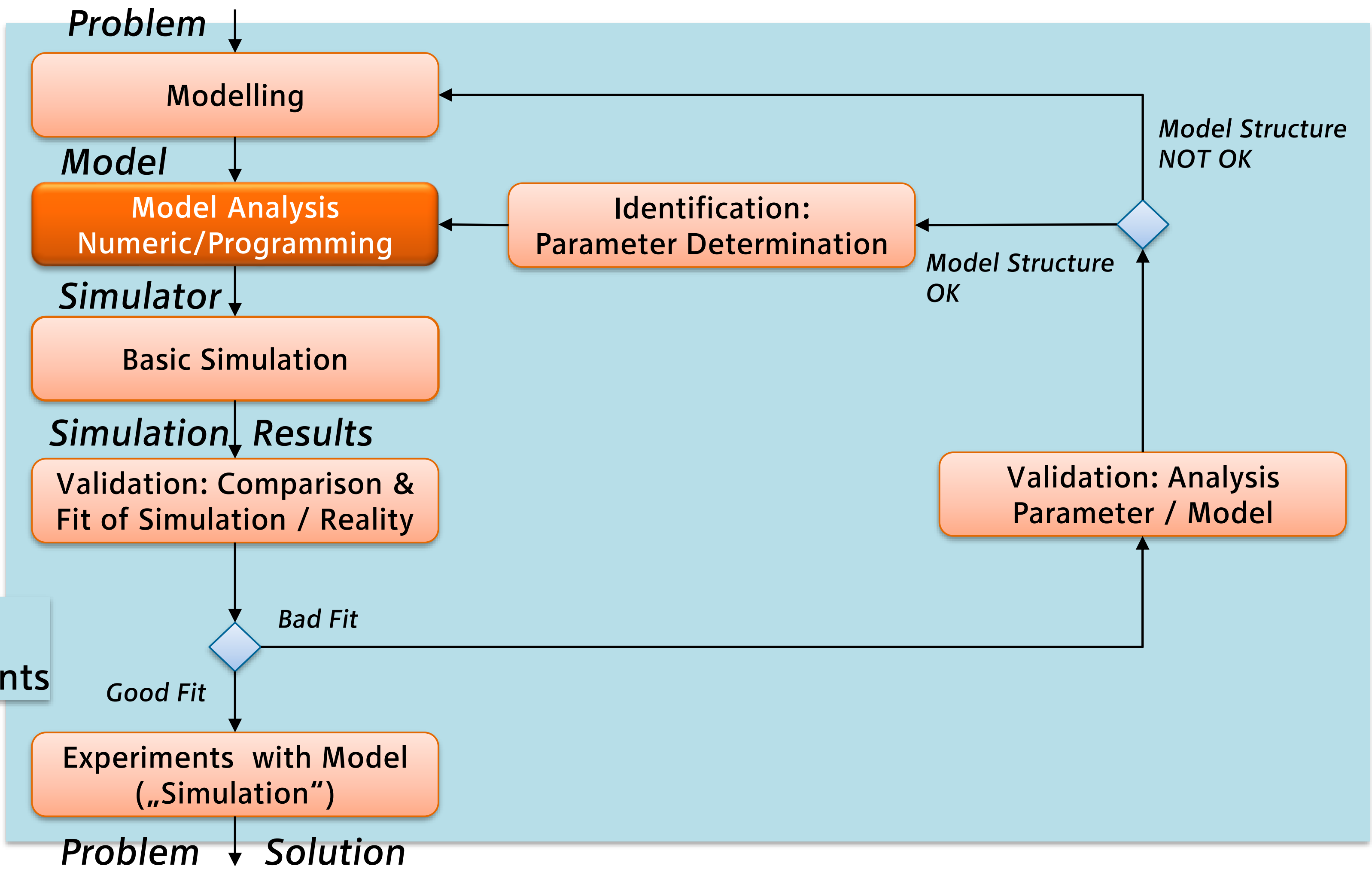
- simulation software
- simulation languages
- Simulators
- Simulation systems
- Simulation Environments



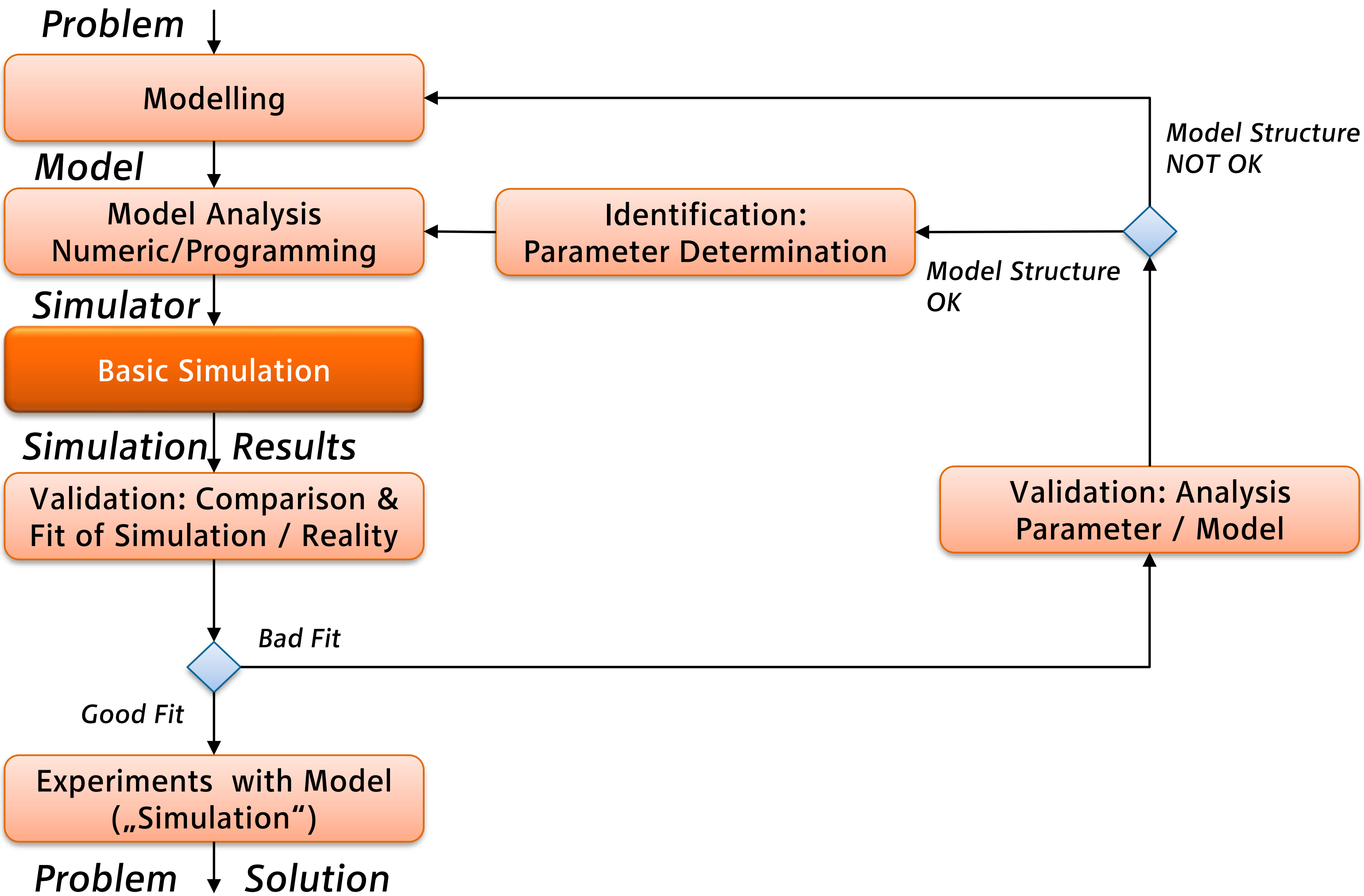
Simulation Circle

Mathematical Modelling and Simulation

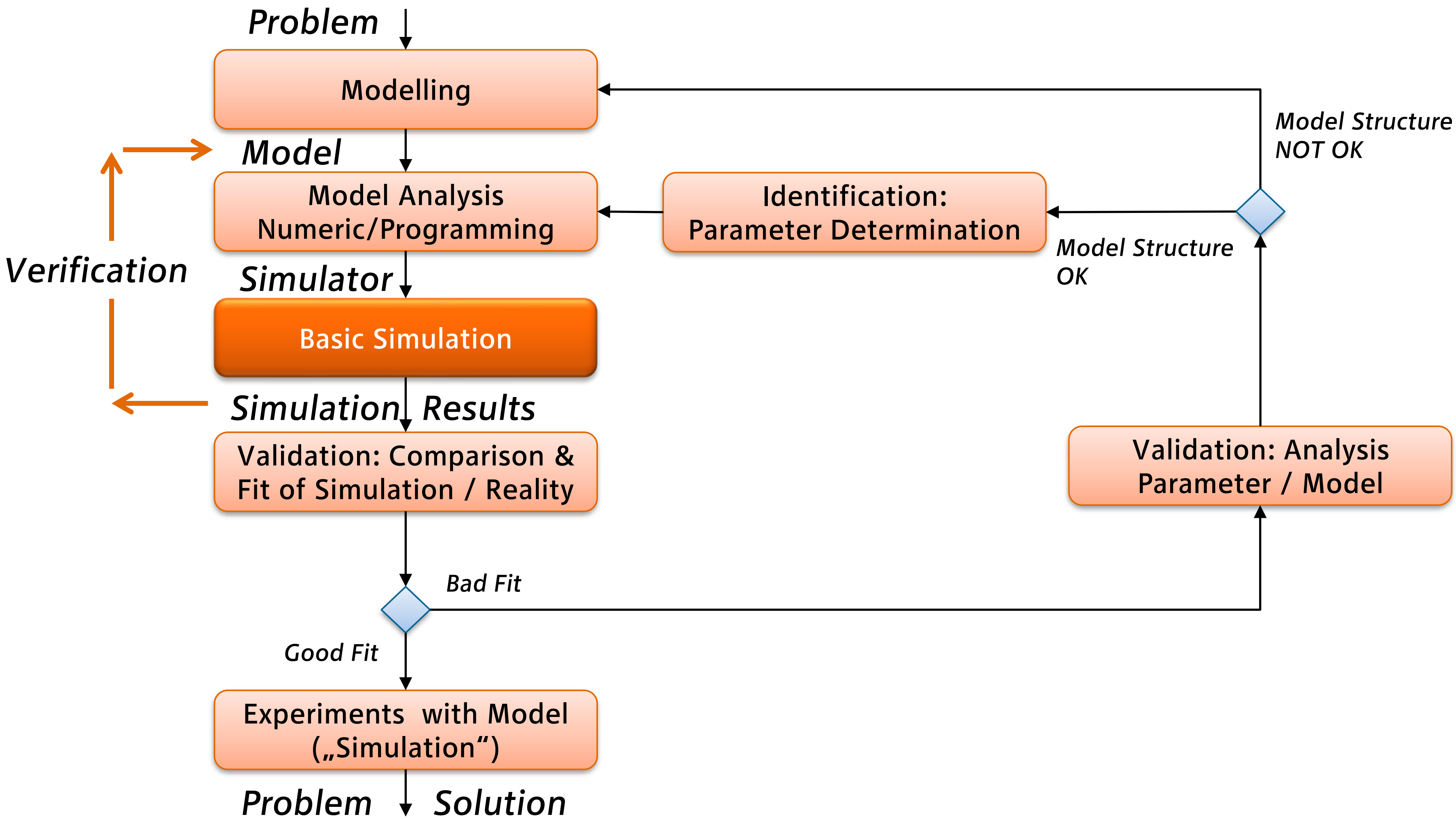
- simulation software
- simulation languages
- Simulators
- Simulation systems
- Simulation Environments



Simulation Circle

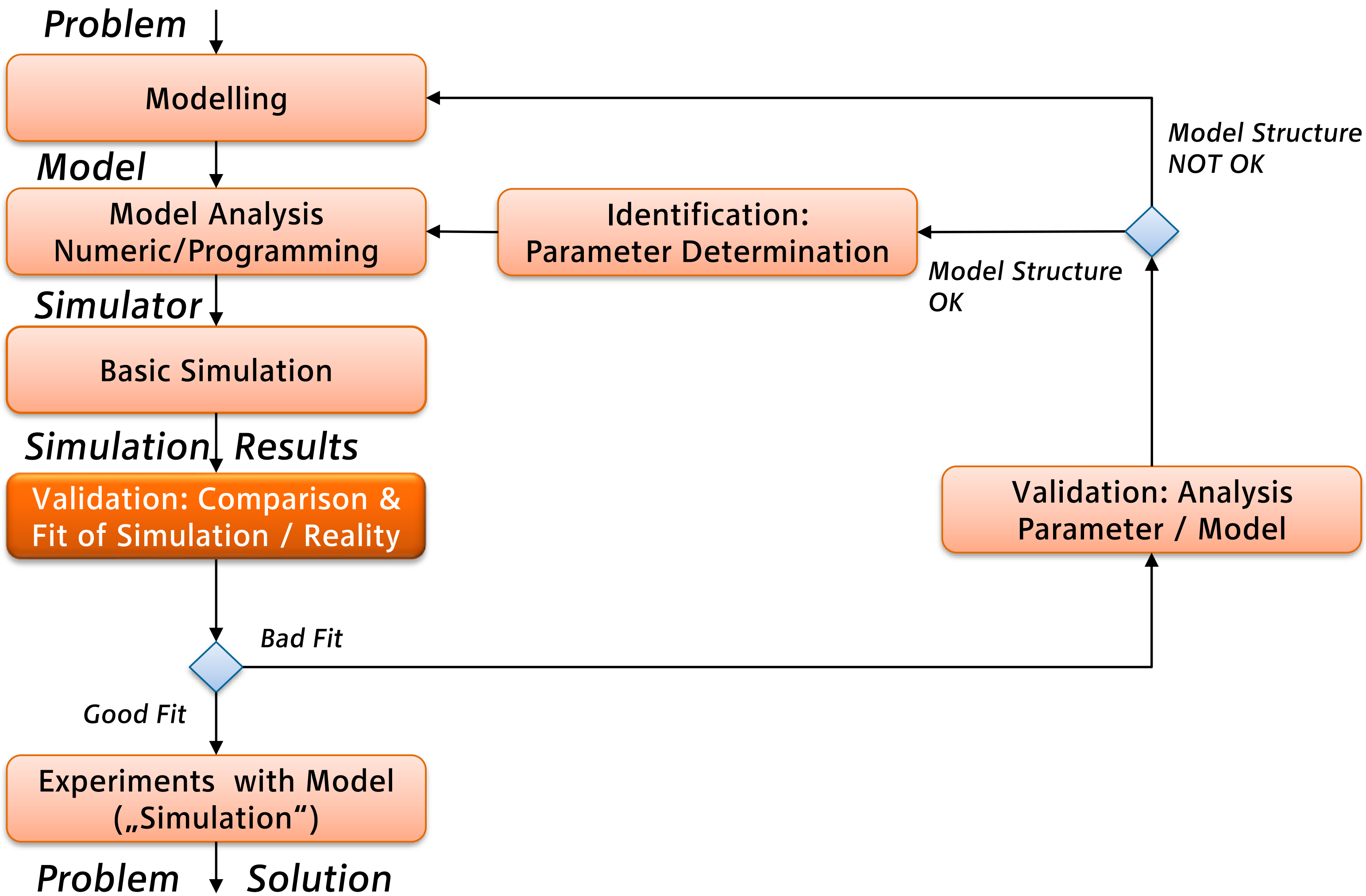


Simulation Circle



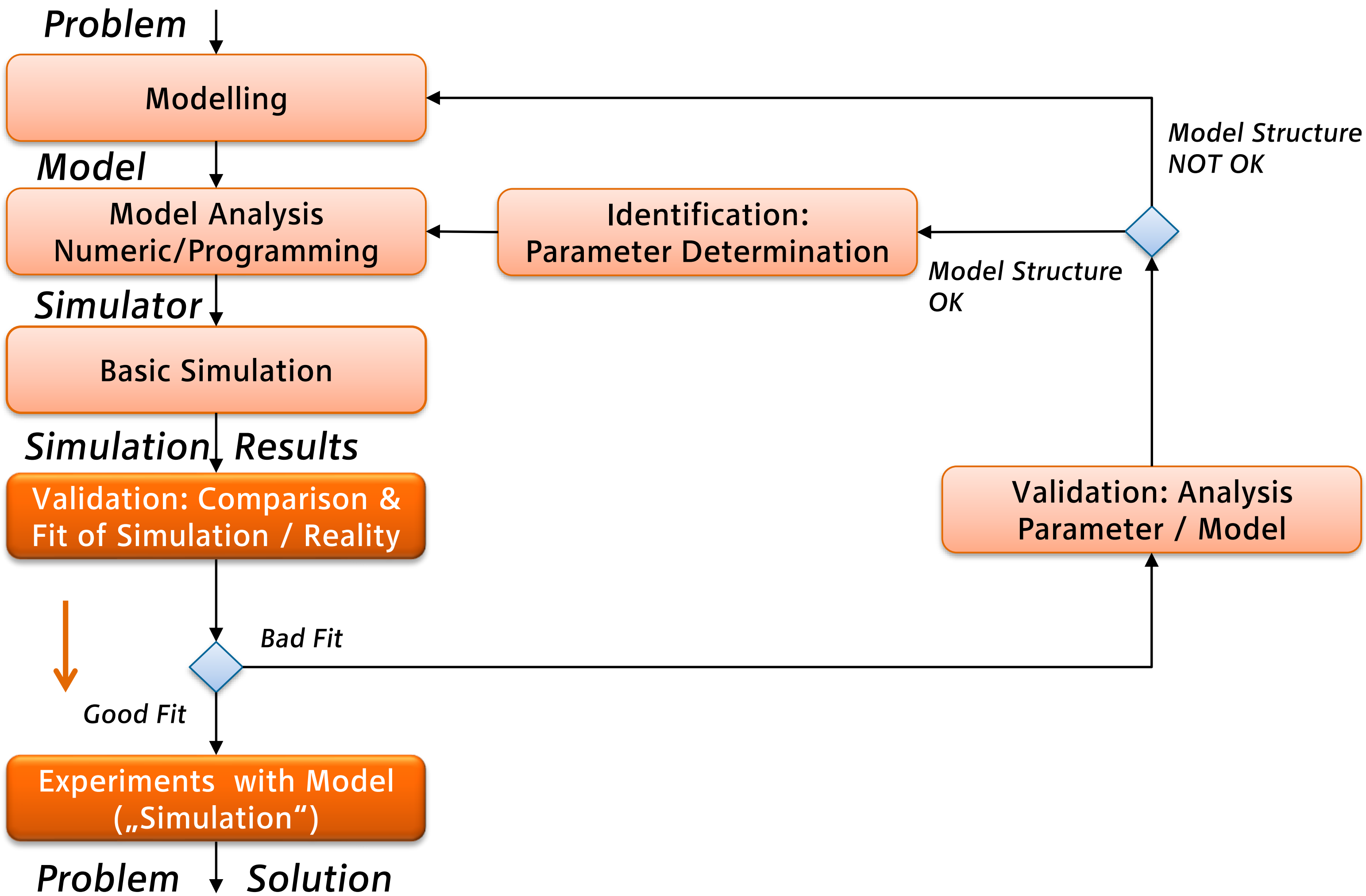
Simulation Circle

Mathematical Modelling and Simulation

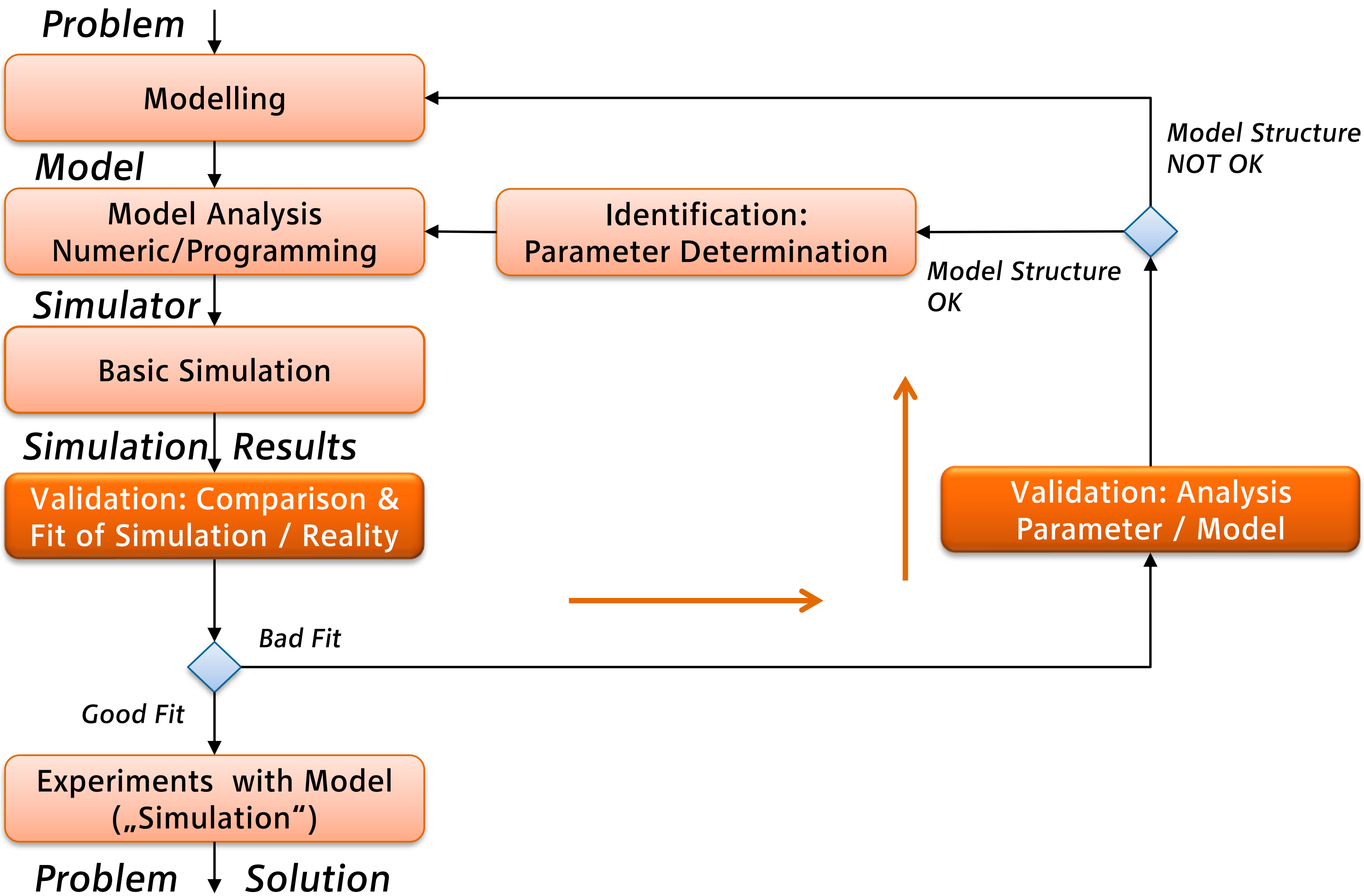


Simulation Circle

Mathematical Modelling and Simulation

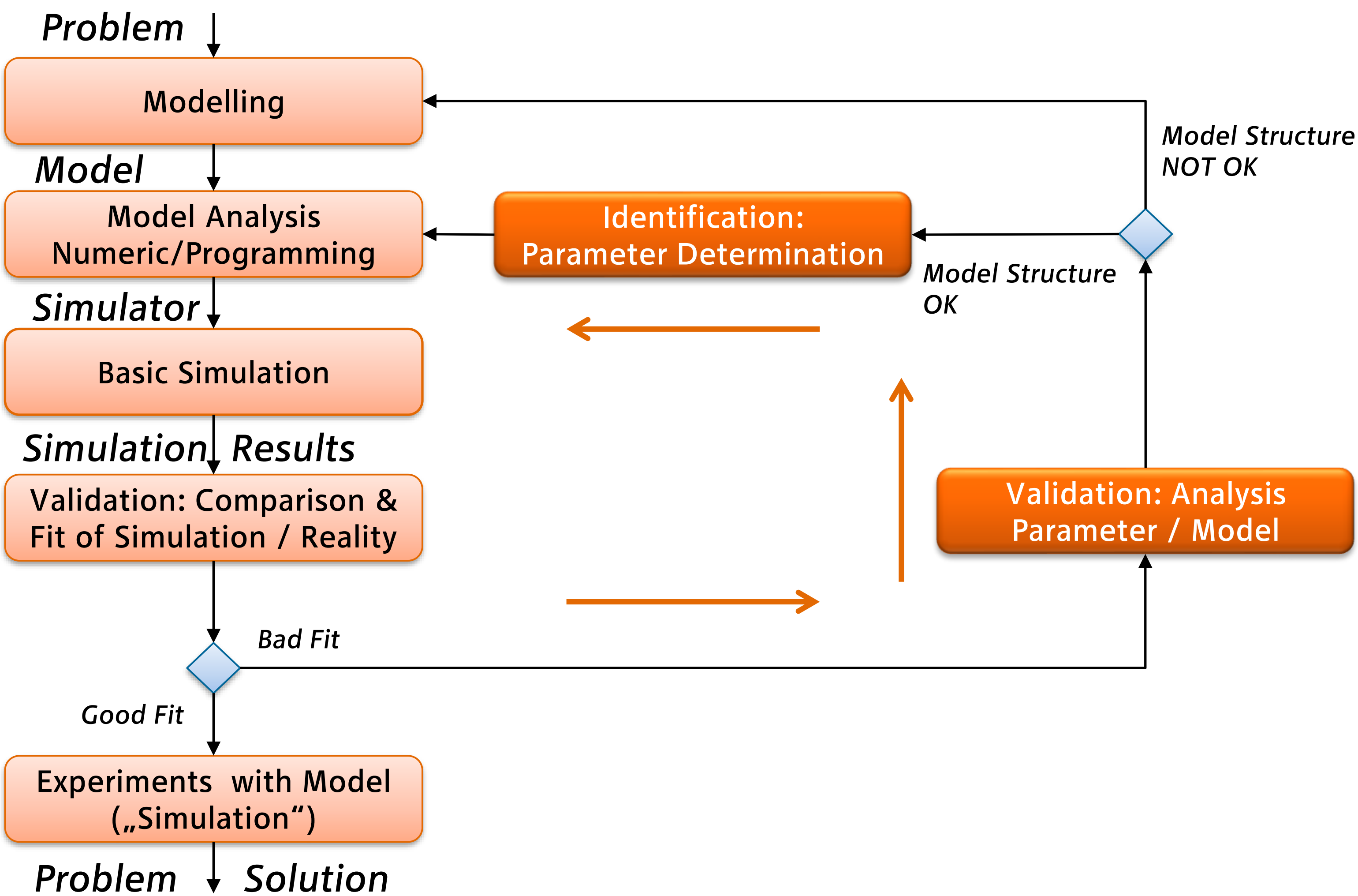


Simulation Circle

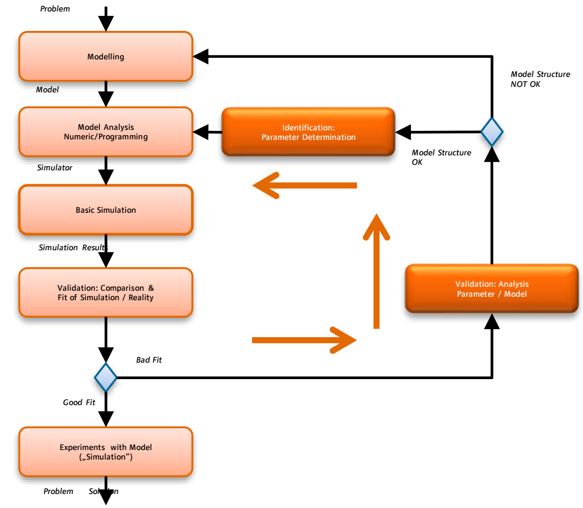


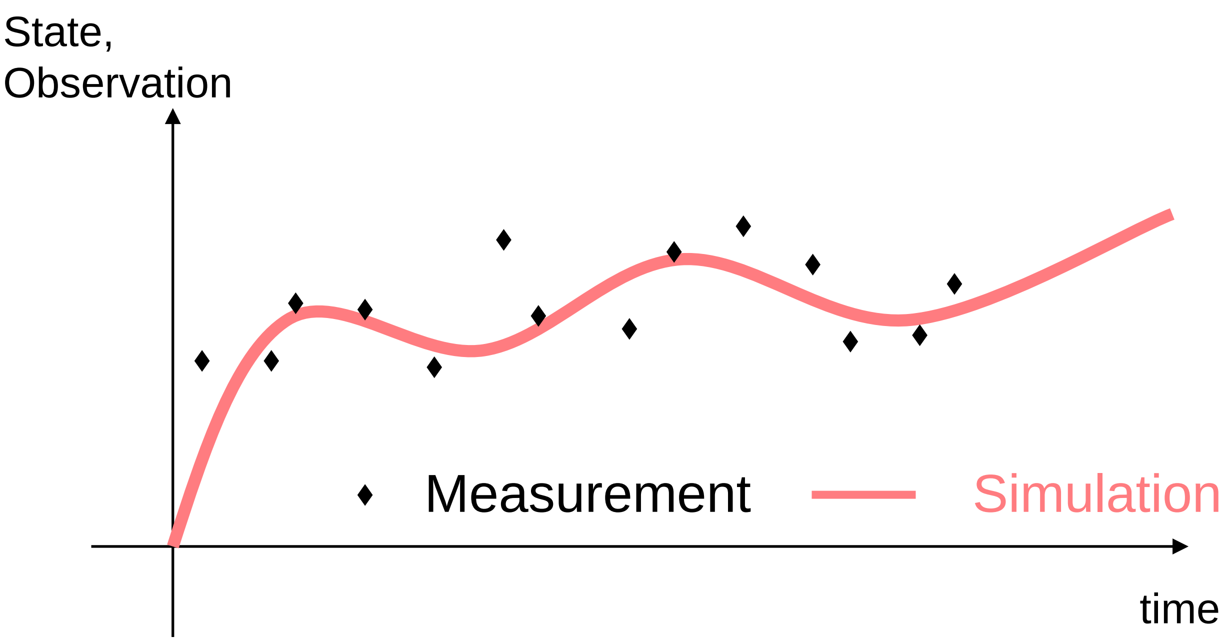
Simulation Circle

Mathematical Modelling and Simulation



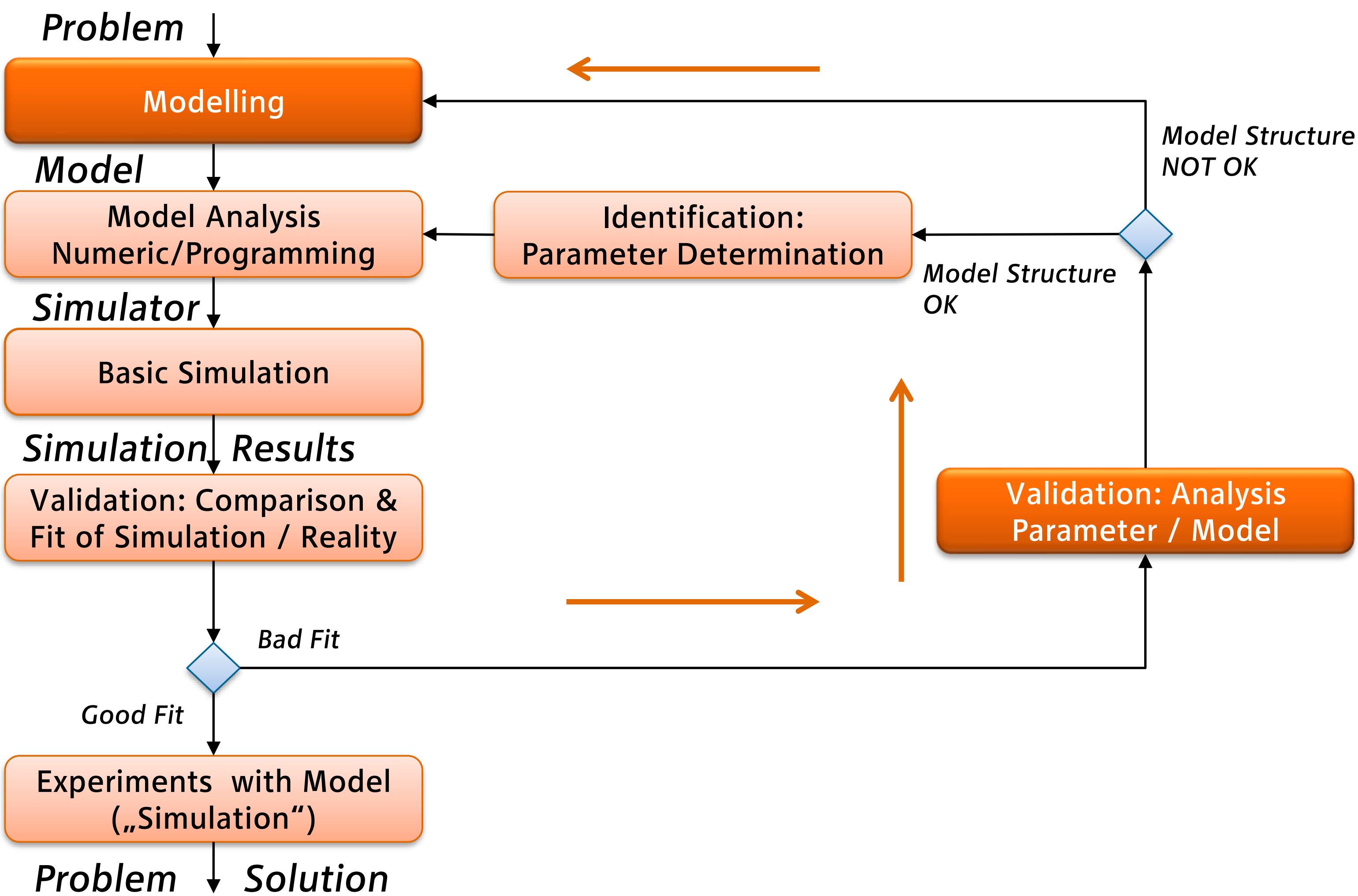
Simulation Circle





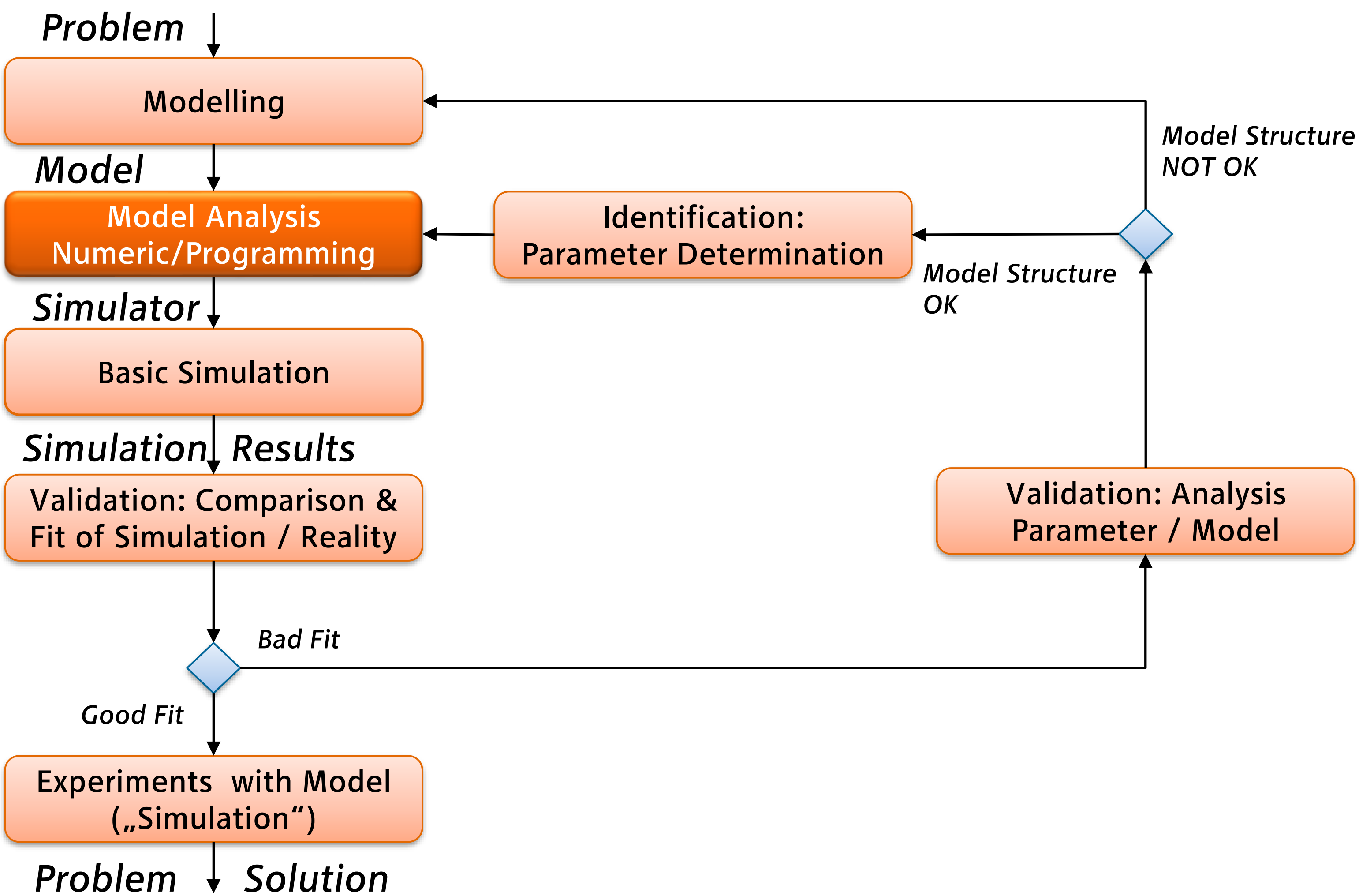
Simulation Circle

Mathematical Modelling and Simulation



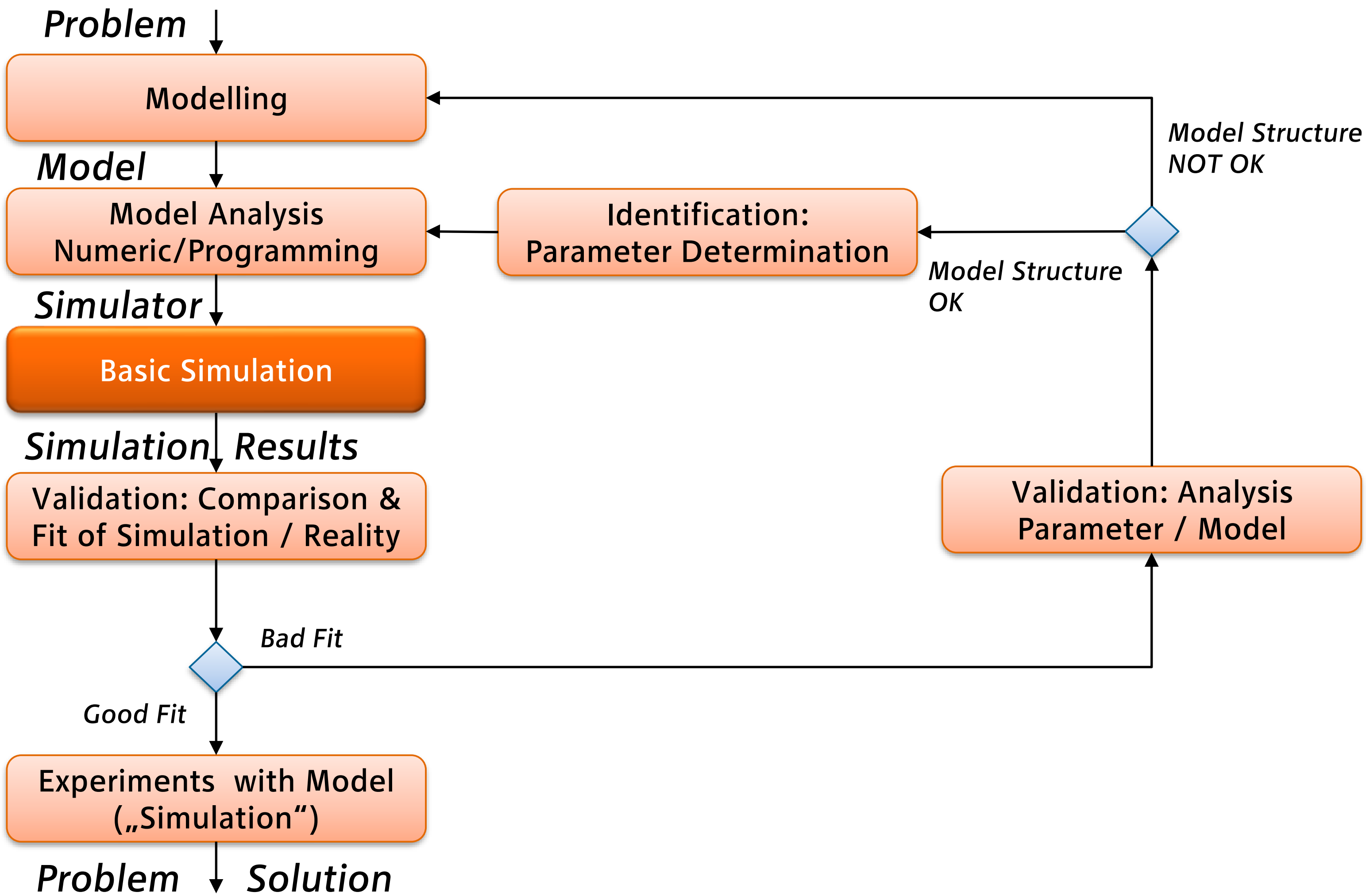
Simulation Circle

Mathematical Modelling and Simulation



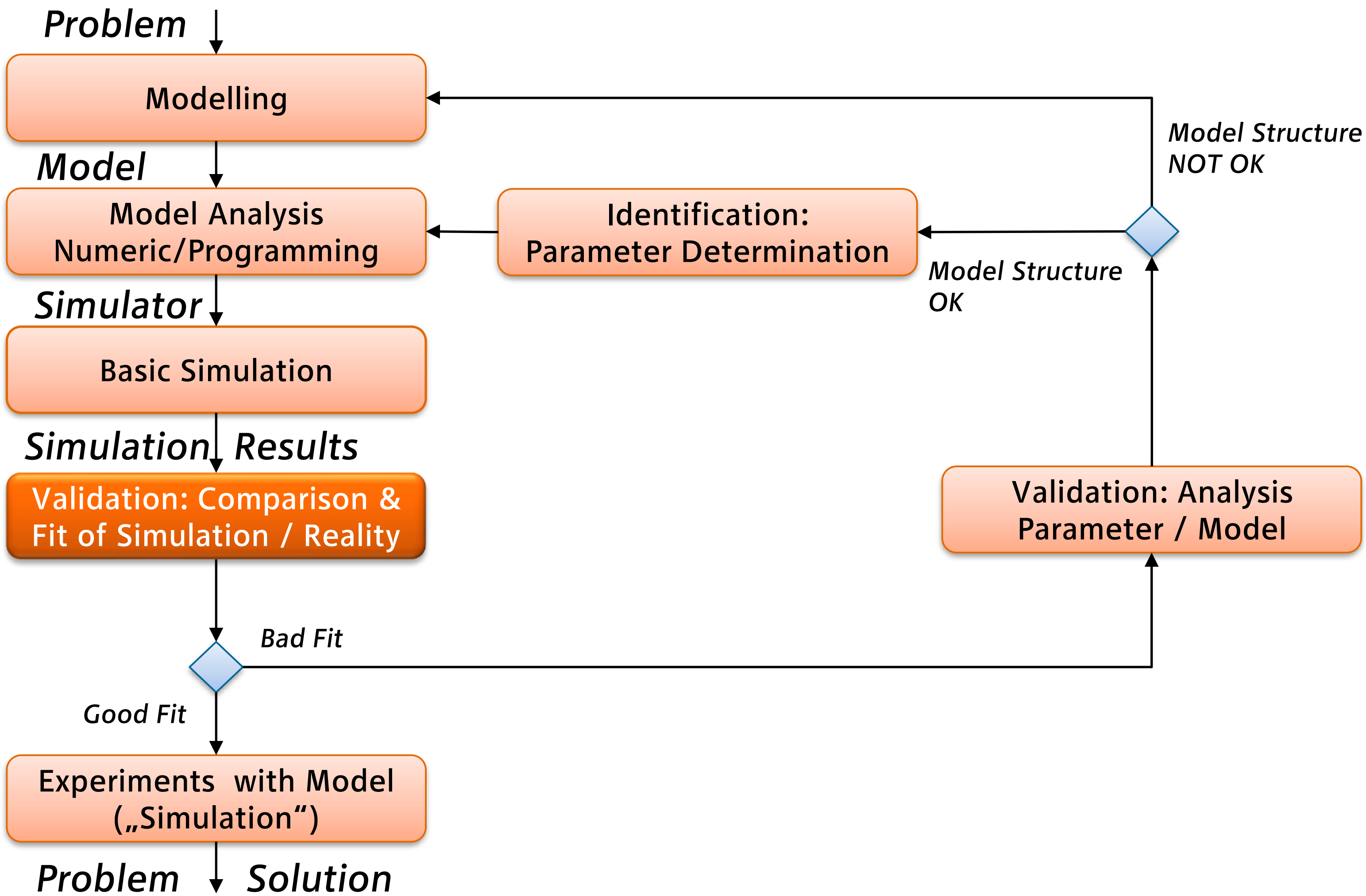
Simulation Circle

Mathematical Modelling and Simulation

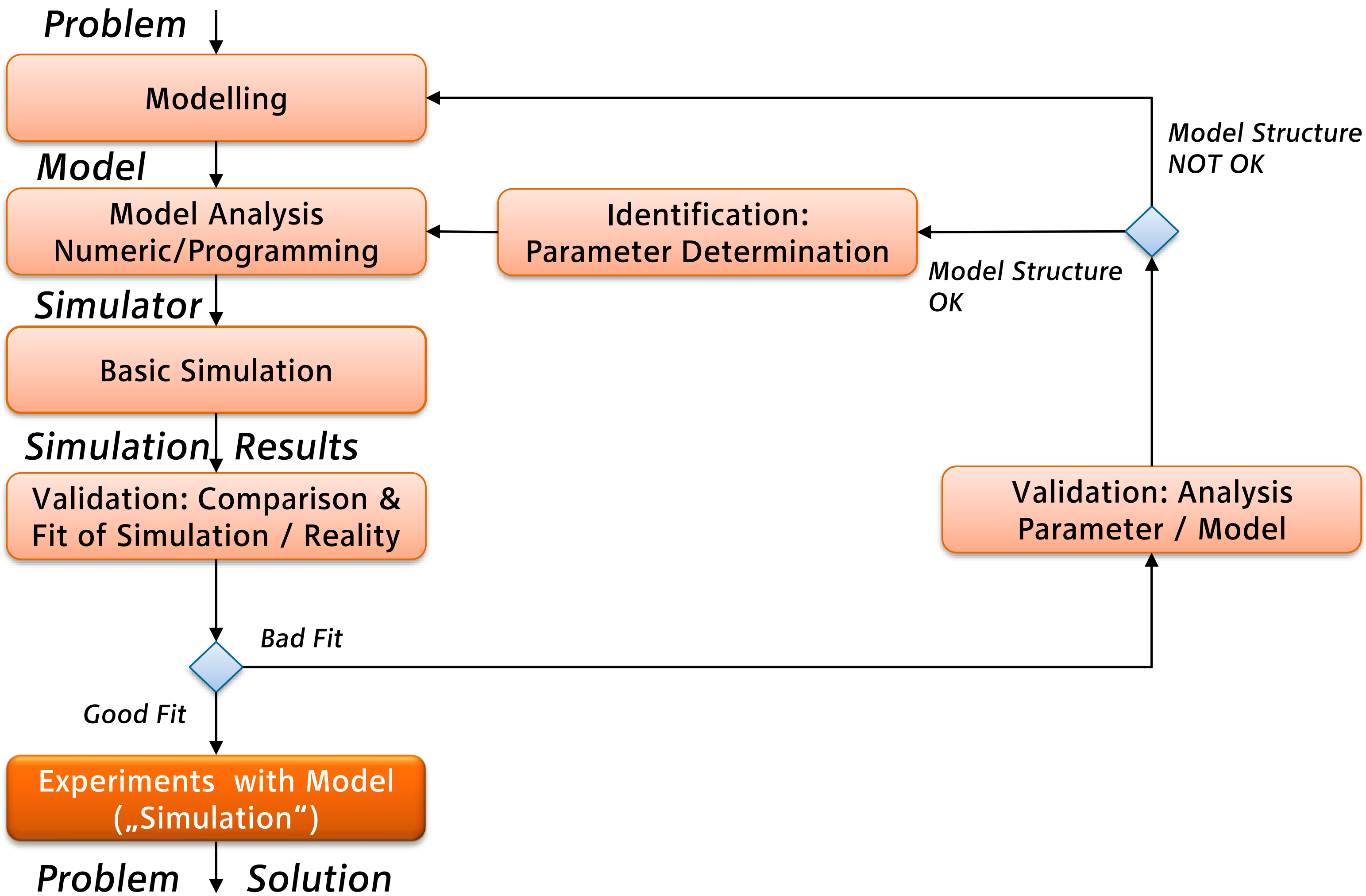


Simulation Circle

Mathematical Modelling and Simulation

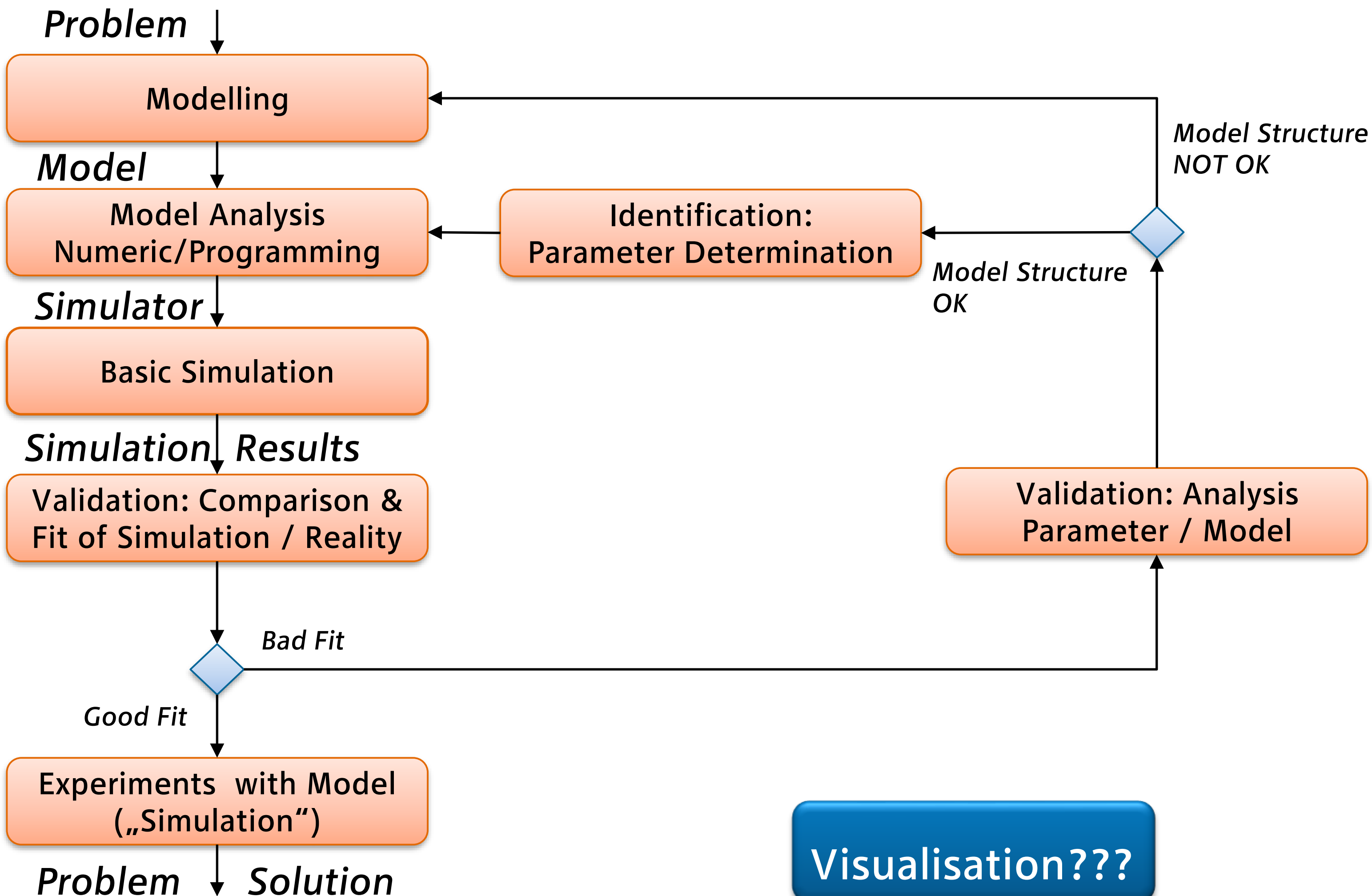


Simulation Circle



Simulation Circle

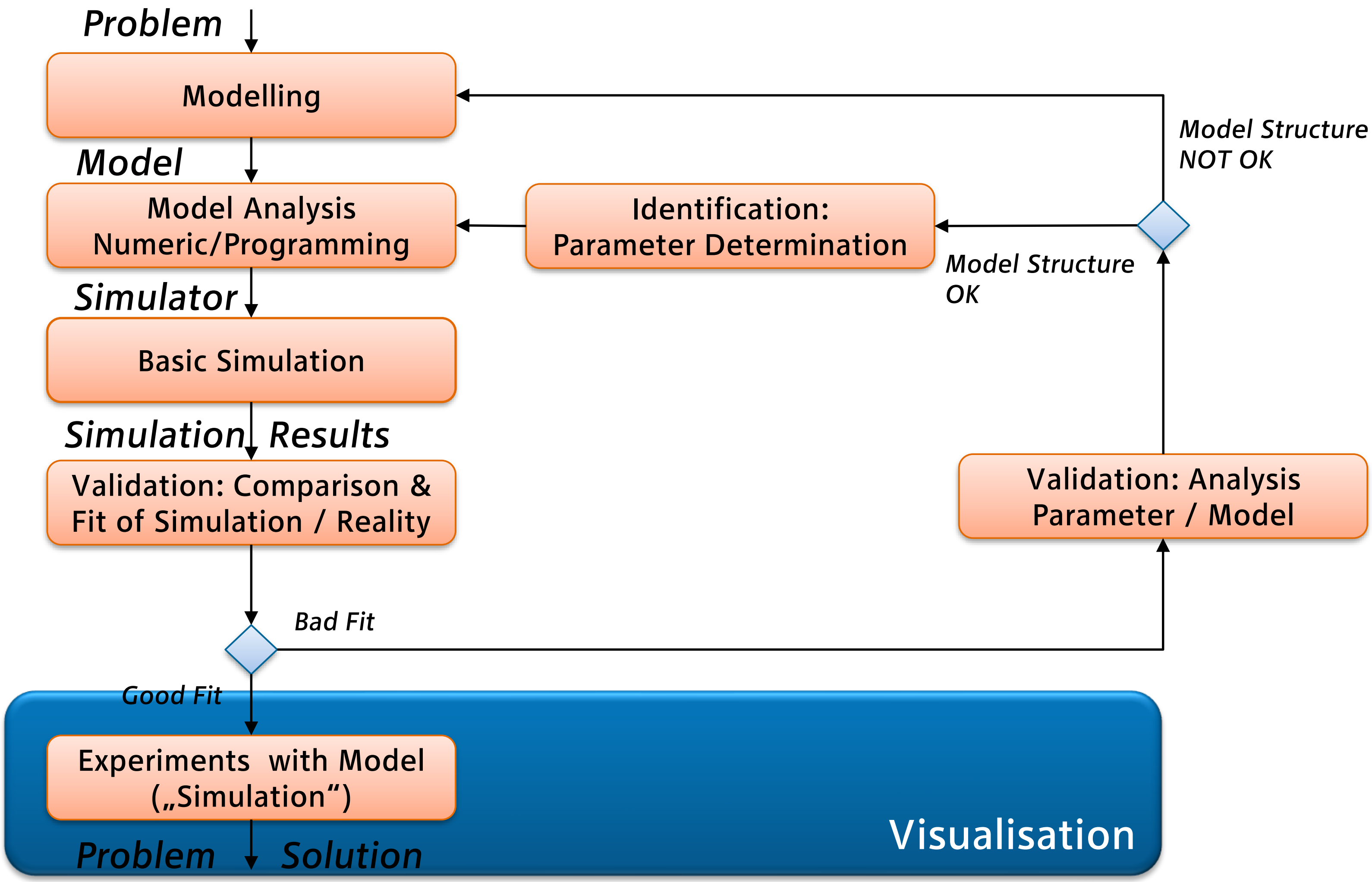
Mathematical Modelling and Simulation



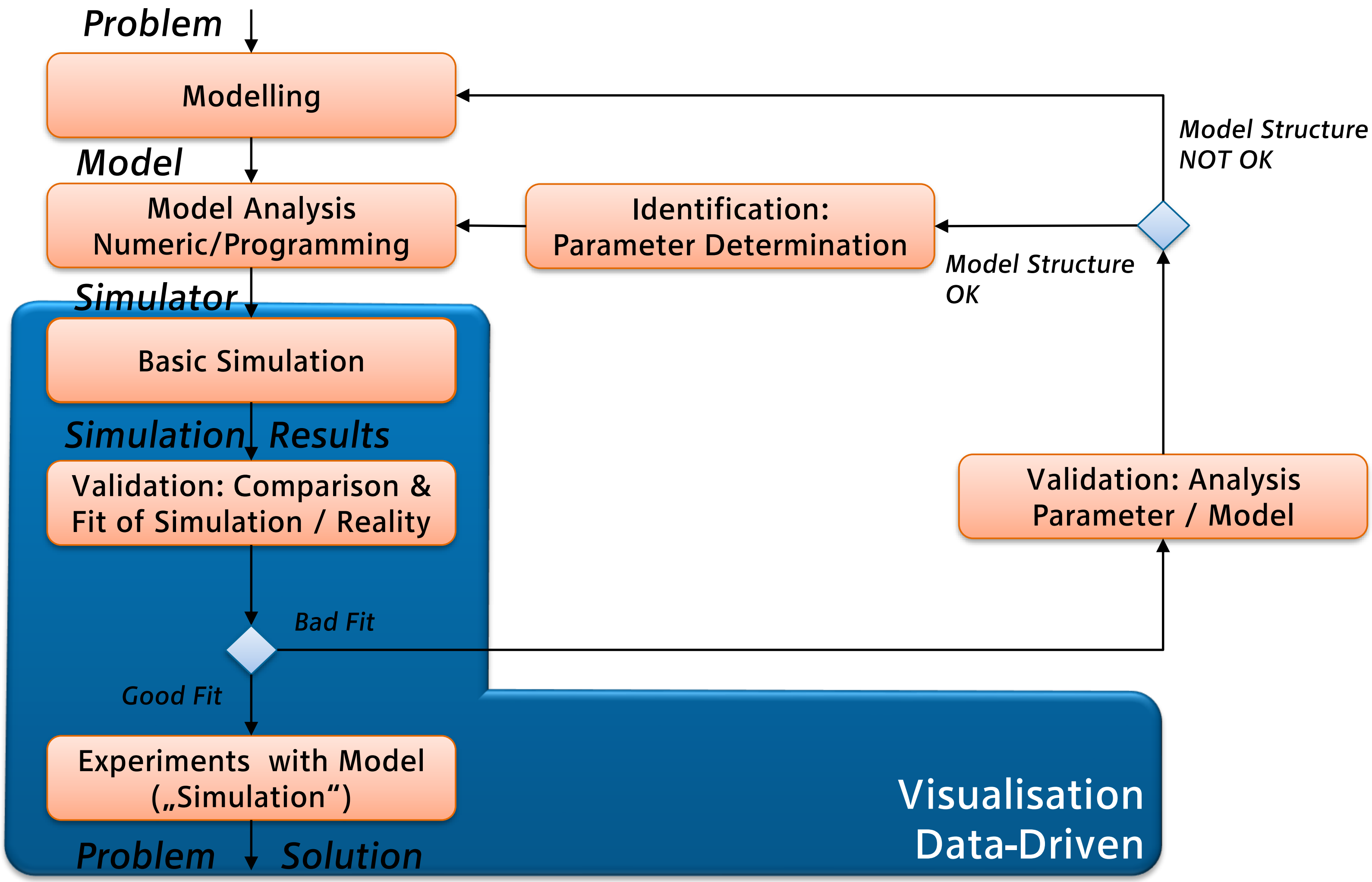
Visualisation???

Simulation Circle

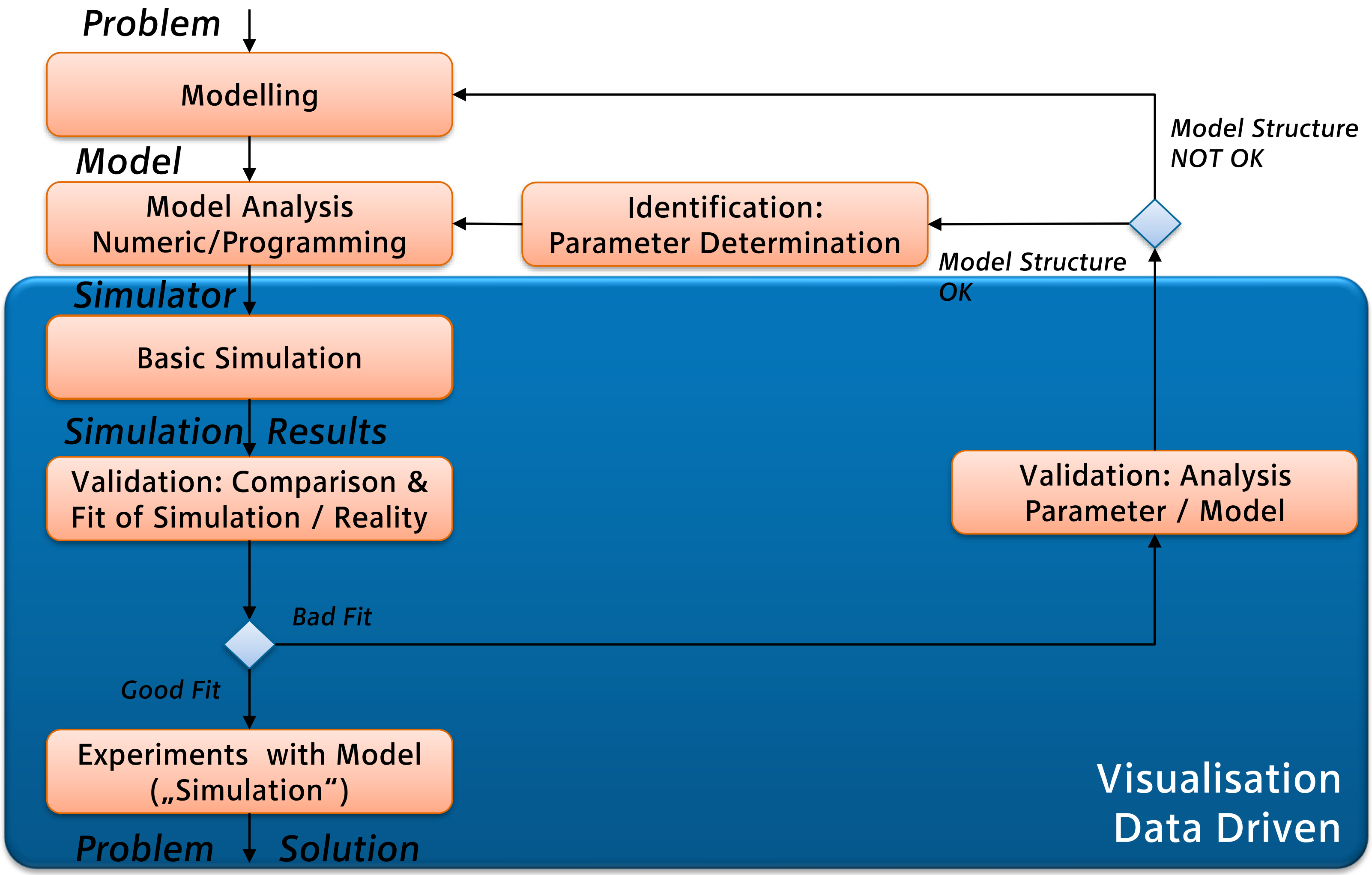
Mathematical Modelling and Simulation



Simulation Circle

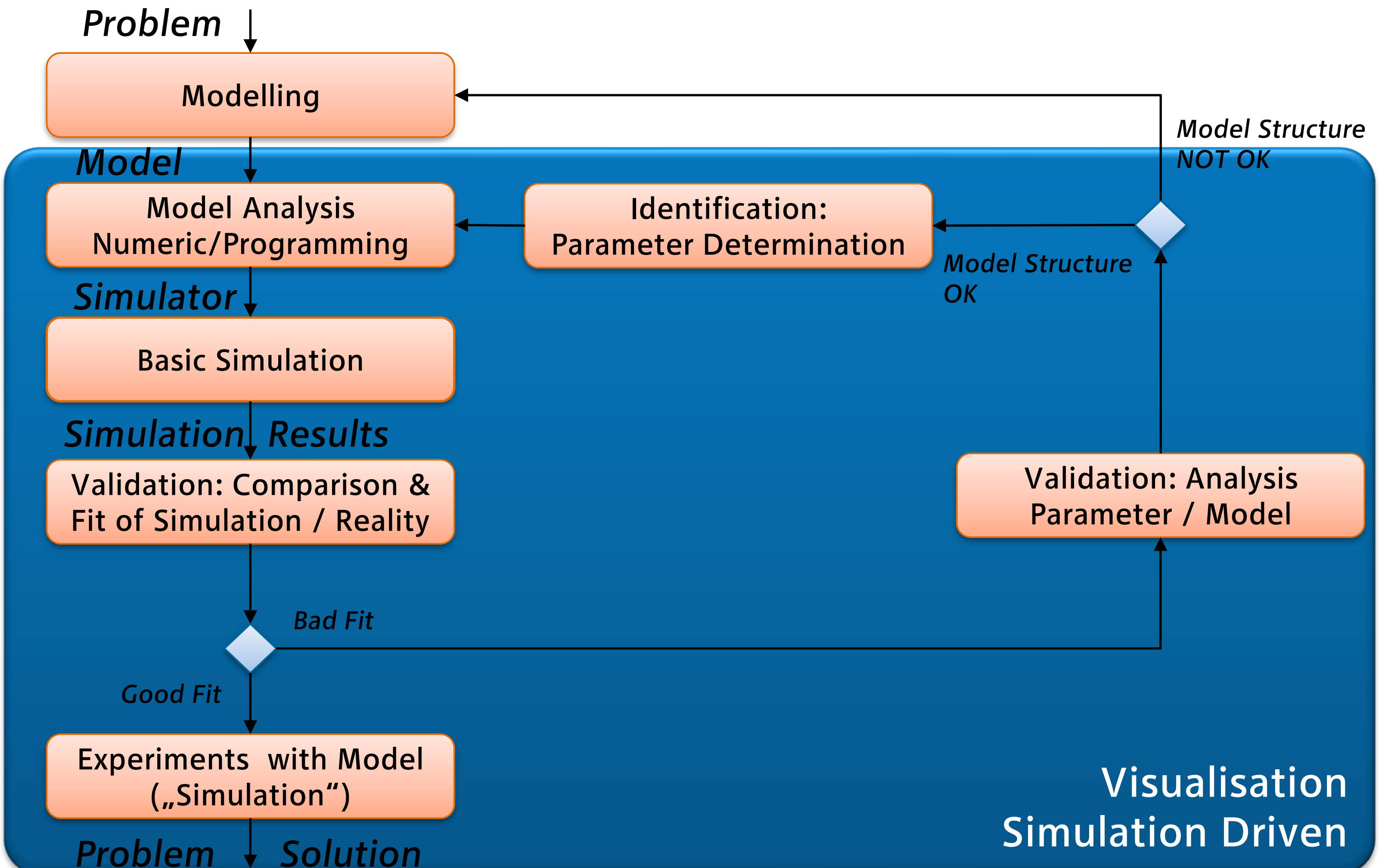


Simulation Circle



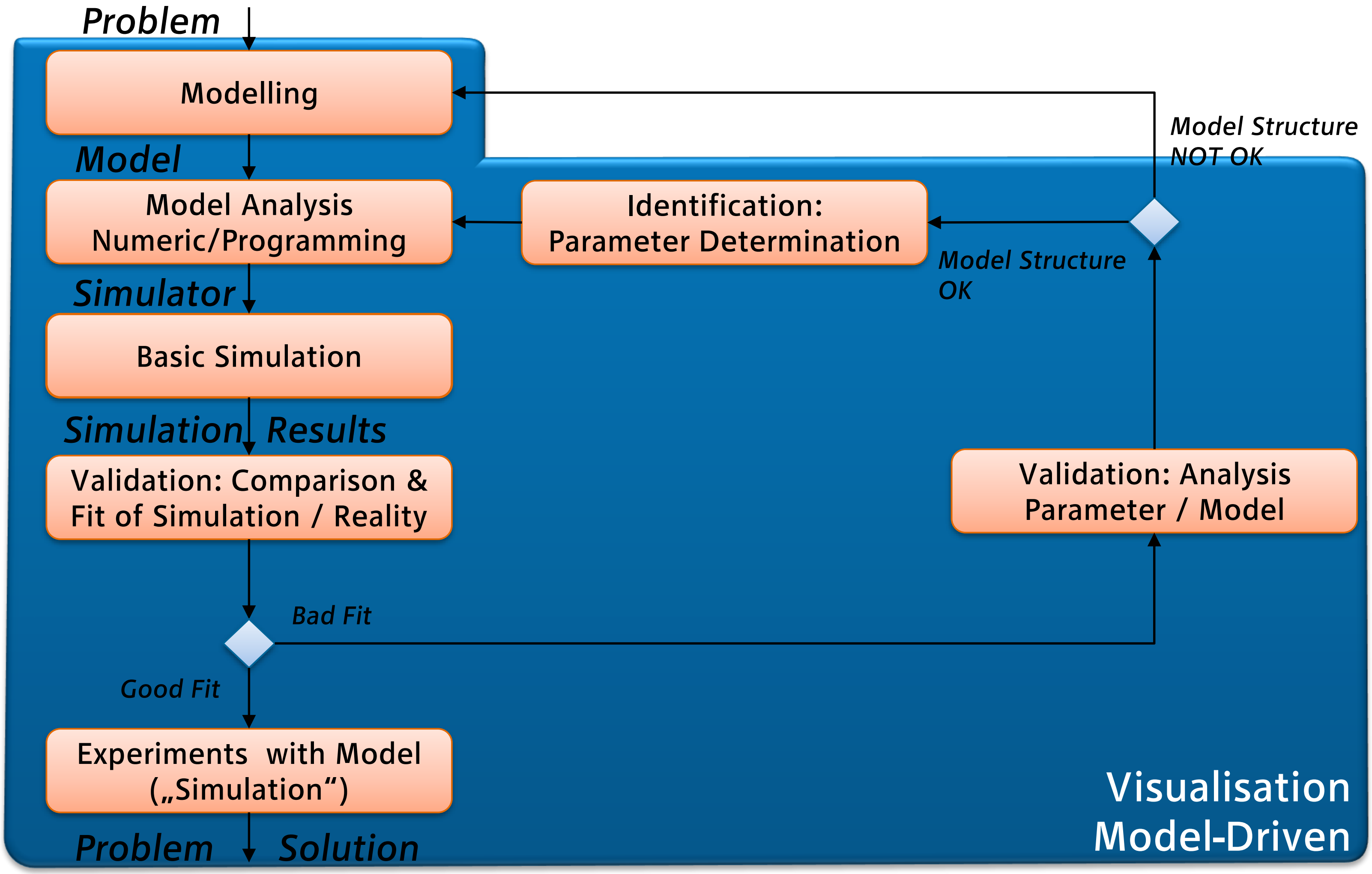
Simulation Circle

Mathematical Modelling and Simulation



Simulation Circle

Mathematical Modelling and Simulation



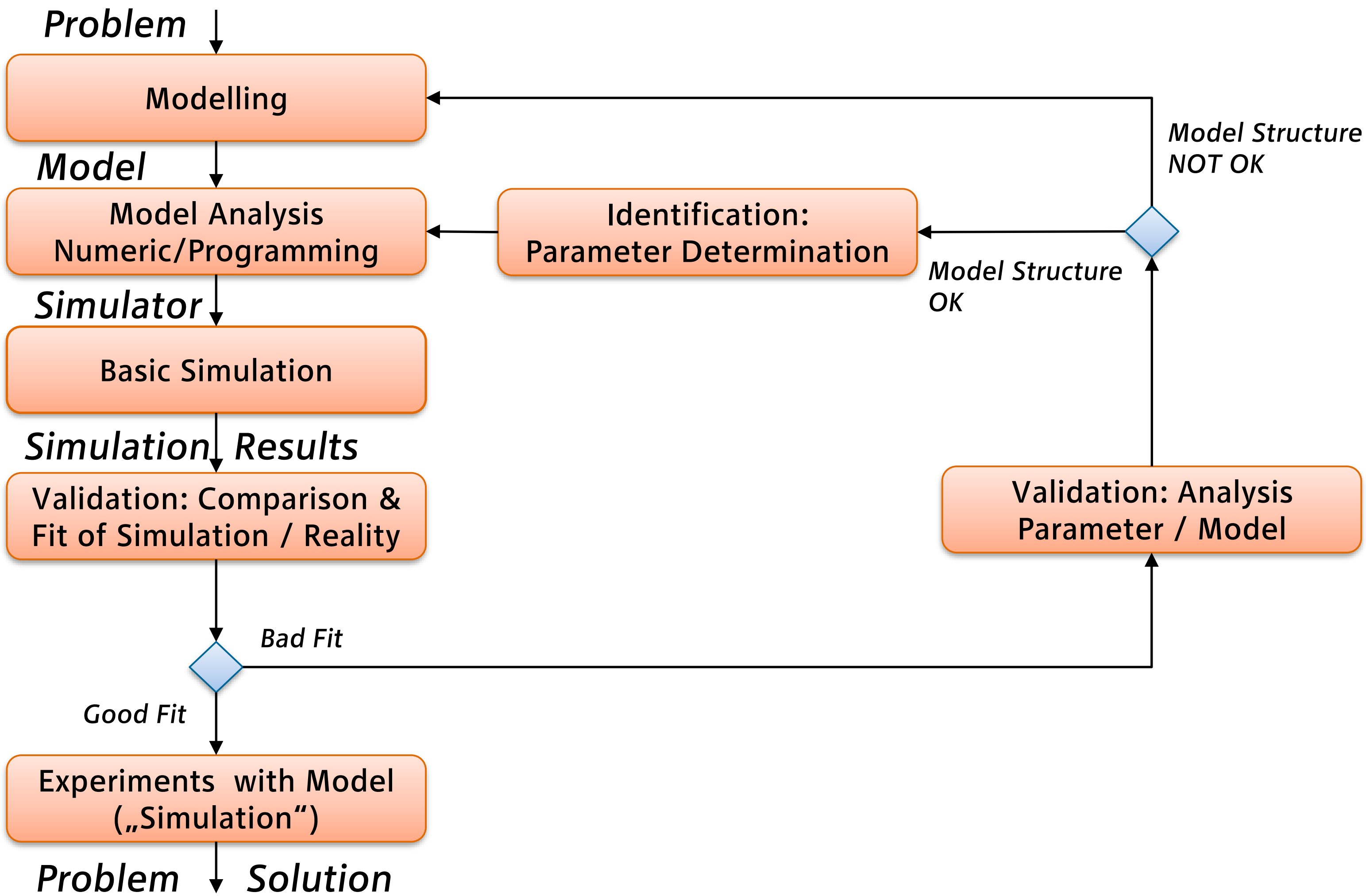
Mathematical Modelling and Simulation

Testcase: Predator-Prey

SIMULATION CIRCLE

Simulation Circle

Mathematical Modelling and Simulation



What is System Dynamics

Mathematical Modelling and Simulation

Forrester, 1961

System Dynamics is a field that resulted from the pioneering efforts of Jay W. Forrester to apply the **engineering principles of feedback and control** to **social systems**.

System Dynamics generates **qualitative models based on causalities**.

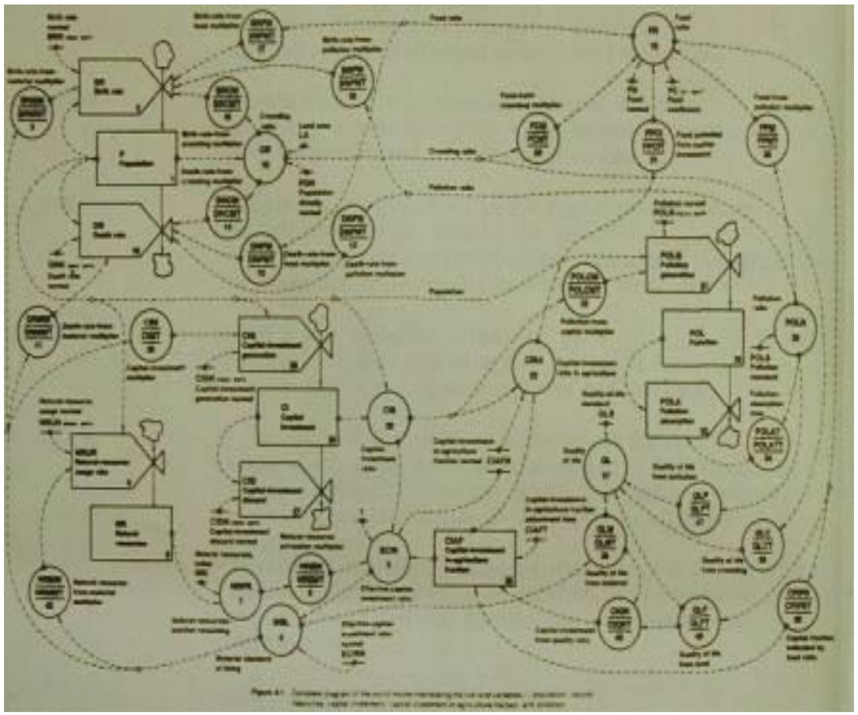
By appropriate parameterisation, the qualitative models can be transformed into “quantitative” **computer models to simulate** the investigated system

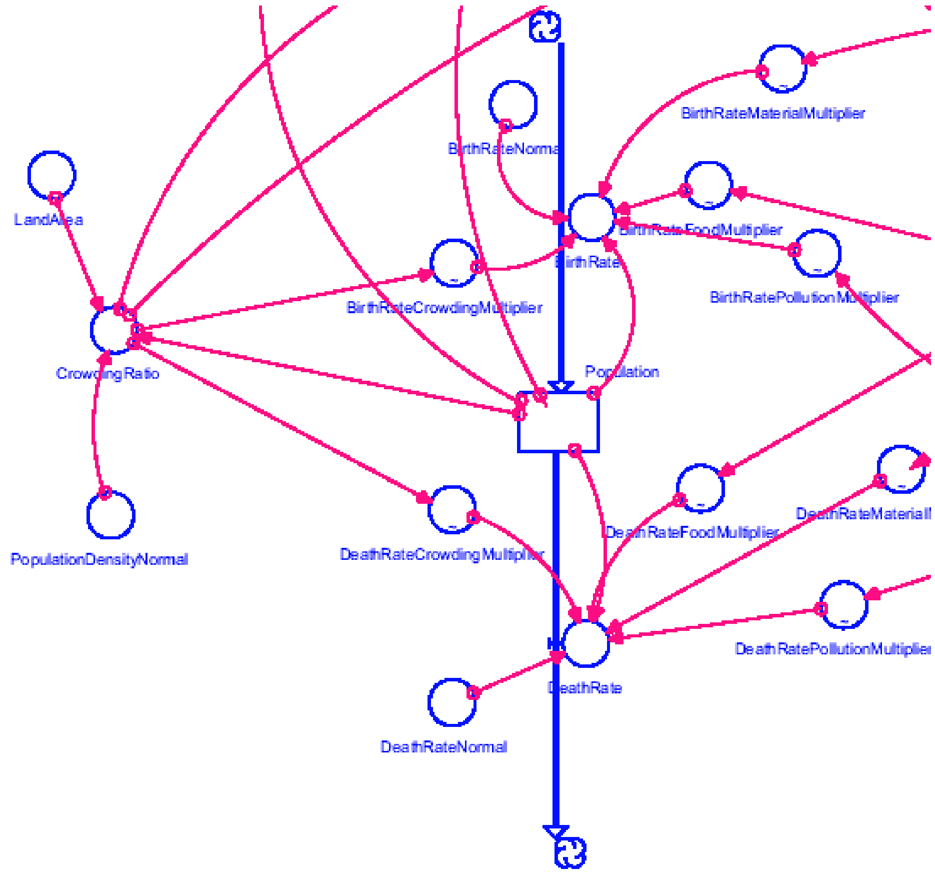
World Models

Systems Dynamics and DYNAMO received widespread interest mainly because they were used to build large world models such as

- WORLD2 (World Dynamics, Forrester1971);
- WORLD3 (The Dynamics of Growth in a Finite World, [Meadows]);
- and WORLD3 revisited (Beyond the Limits).

World Models

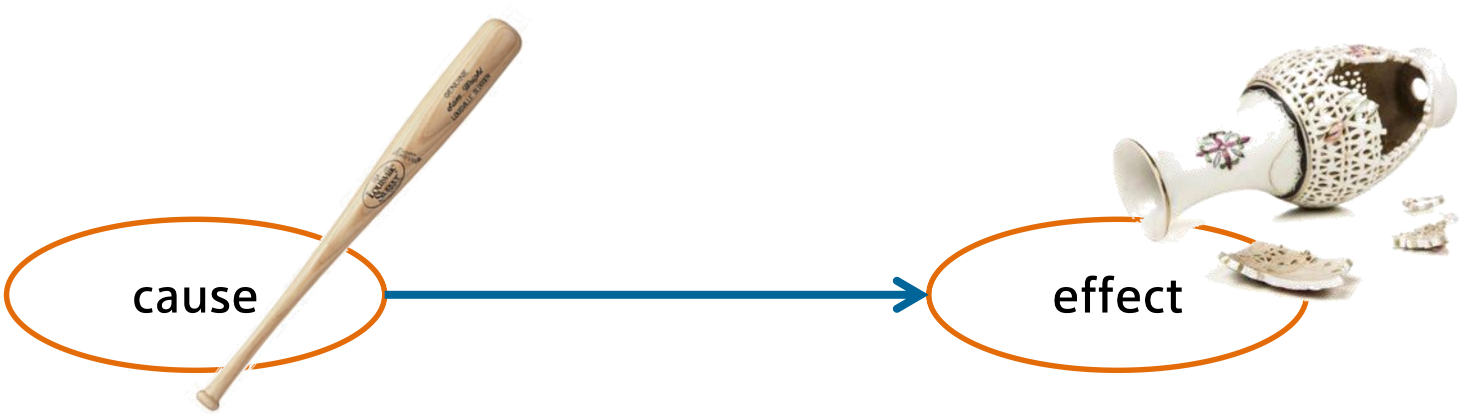




Key to develop SD Models

Causal thinking is the key to organizing ideas in a system dynamics study

(Roberts et al. 1983)



How to build a SD Model?

1. Identify system variables and system boundaries
2. Capture **links** of variables in a **Causal Loop Diagram (CLD)**
3. Build a **Stock and Flow Diagram (SFD)**

- Implement the model in a simulator

1. System Variables and Boundaries

Mathematical
Modelling and
Simulation

a. Analysis of the problem - Determining the purpose and the use of the model and defining a target for the simulation.

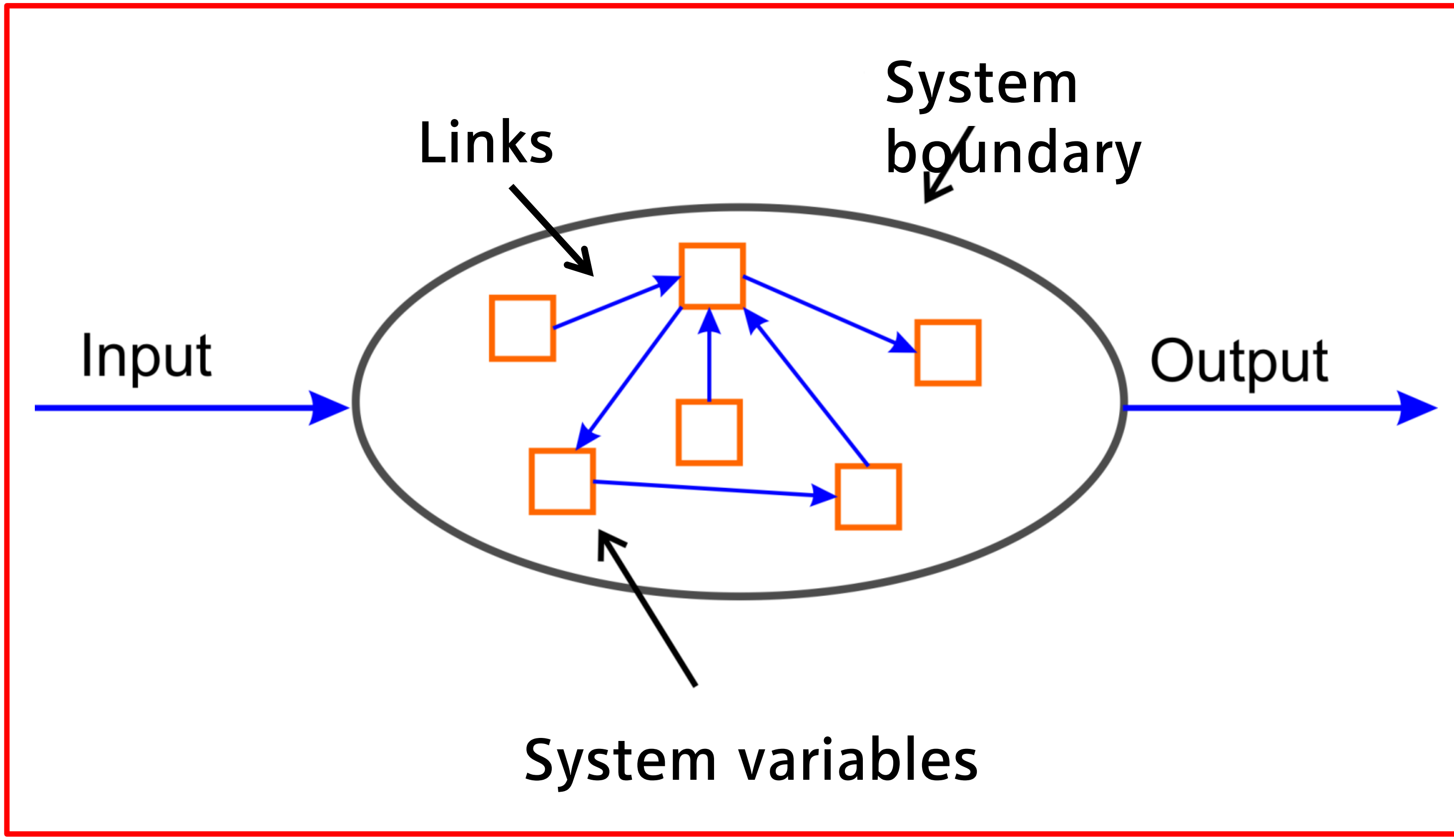
b. Start collecting information and data. Start developing hypothesis about the parts of the system.

c. Determine the elements of the system.

d. Determine causal relationships between the elements.

1. System Variables and Boundaries

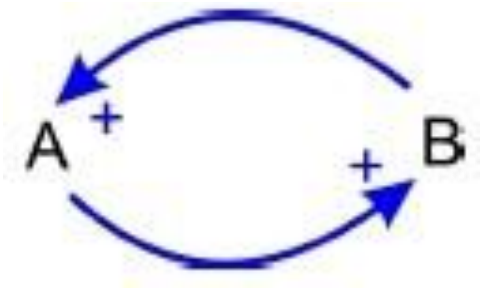
Mathematical Modelling and Simulation



2. Causal Loop Diagram

Mathematical Modelling and Simulation

Capture the **behavior** and **links** of and within the system by interlinking system variables that are related to each other



<u>Behavior of system due to</u>:

- Feedback Loops
- System memory (stocks)
- Delays in material and information delays

2. Causal Loop Diagram

<u>Main components of CLDs:</u>

- **System variables:** names of elements
- **Link - positive:**

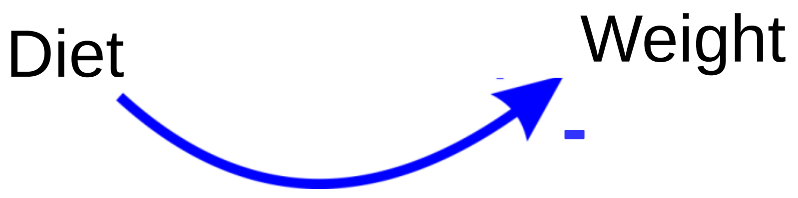
Represented by a plus-sign

Increase in variable *Eating* results in an increase in variable *Weight*

2. Causal Loop Diagram

Main components of CLDs:

- **Link – negative:**



Represented by minus-sign.

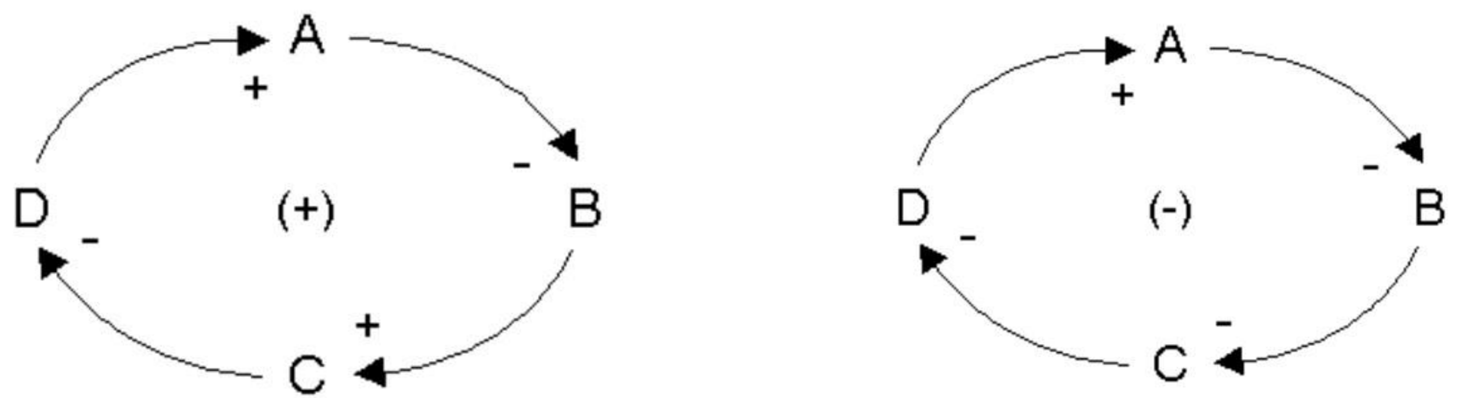
Increase in variable *Diet* results in a decrease in variable *Weight*

2. Causal Loop Diagram

Mathematical Modelling and Simulation

Main components of CLDs:

- **Feedback Loops:** are closed loops of arrows, represented by a:
“(+)” (or ”(R)” for **reinforcing**) or
“(**-**)” (or “(B)” for **balancing**) sign in the middle.

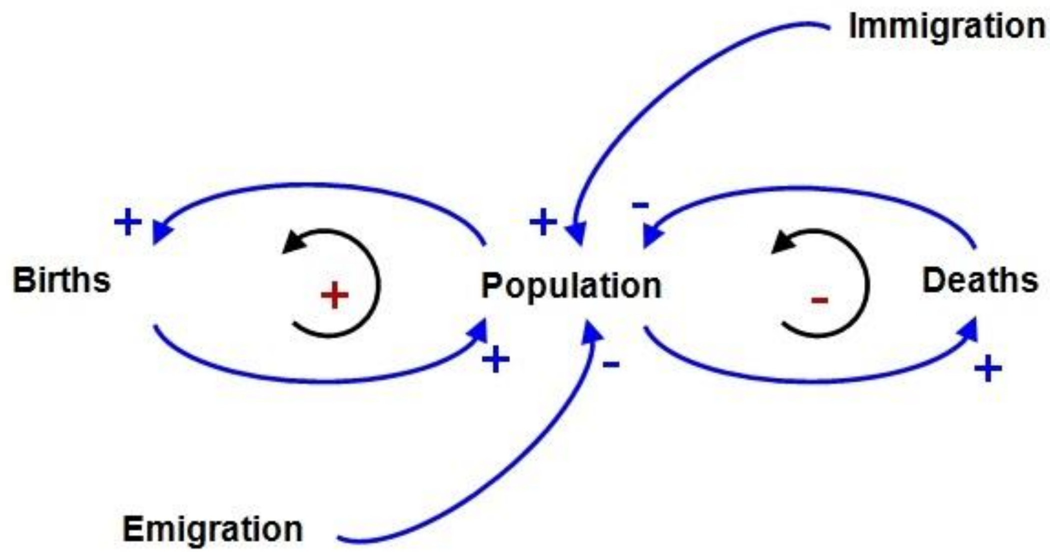


2. Causal Loop Diagram

Mathematical Modelling and Simulation

<u>Main components of CLDs:</u>

- **Feedback Loops:** are closed loops of arrows, represented by a
“(+)” (or ”(R)” for **reinforcing**) or
“(-)” (or “(B)” for **balancing**) sign in the middle.



TU WIEN Mathematical Modelling and Simulation

Feedback Loops

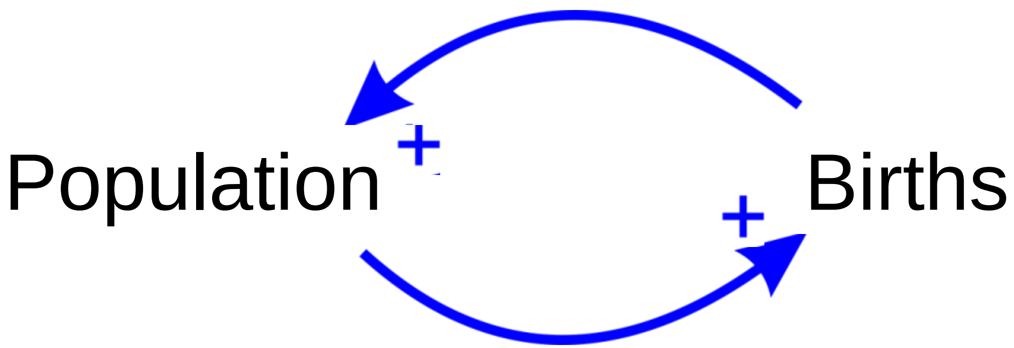
- Search to identify closed, causal feedback loops is one key element of System Dynamics
- The most important causal influences will be exactly those that are enclosed within feedback loops

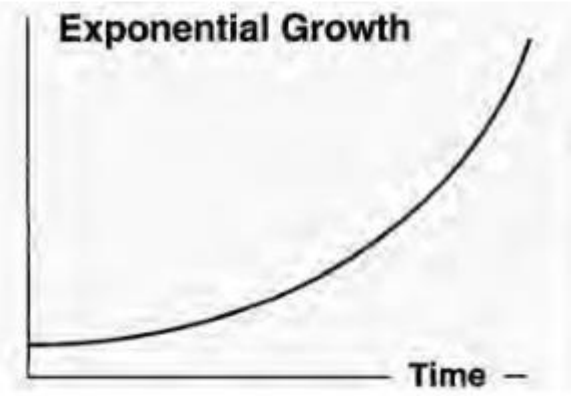
2. Causal Loop Diagram

<u>Types of behavior due to loops:</u>

- **Exponential Growth**: arises from **positive (reinforcing) feedback** loop.

Example:



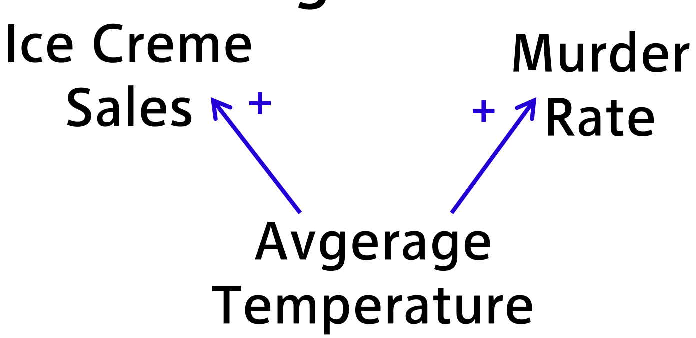


<u>Causation vs. Correlation</u>

- ***Correlation*** represents past behavior and not the structure of the system
- ***Causation*** represents the causal links of the structure

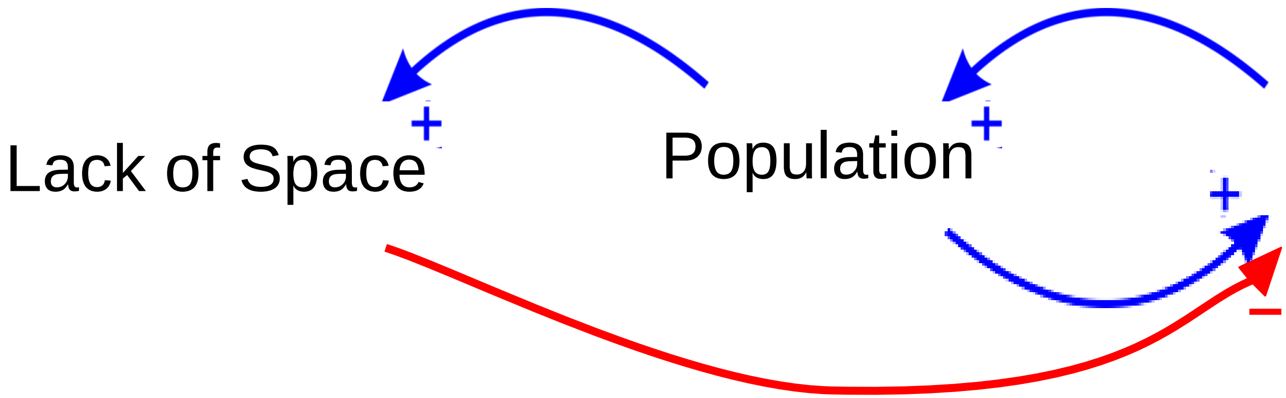
Wrong:

Right:



2. Causal Loop Diagram

At least one negative feedback loop is necessary to receive a stable system



3. Stock and Flow Diagram

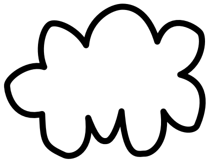
Mathematical Modelling and Simulation

Problem: Not all system elements are system variables!

Solution: distinguish between

- Sources/Sinks
- Levels/Stocks
- Flows
- Auxiliaries
- Paramters
- Links

Sources/Sinks:



Source represents systems of levels and rates outside the boundary of the model

Sink is where flows terminate outside the system

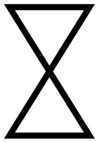
E.g.: Raw Material (Source for „Construction“ Flow), Graveyard (Sink for „Dying“ Flow)

Levels/Stocks/System variables:

A quantity that accumulates over time and changes its value continuously.

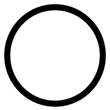
E.g.: Size of a population, Number of people waiting in a queue, Number of goods waiting to be transported, etc.

Flow/Rate/Activity/Movement:



Changes the values of levels. Every level has at least to be connected to one flow in order to change its value.

E.g.: Birth (Changes the value of the stock „population"), Eating (Changes the value of the stock „amount of food"), etc.



Auxiliary:

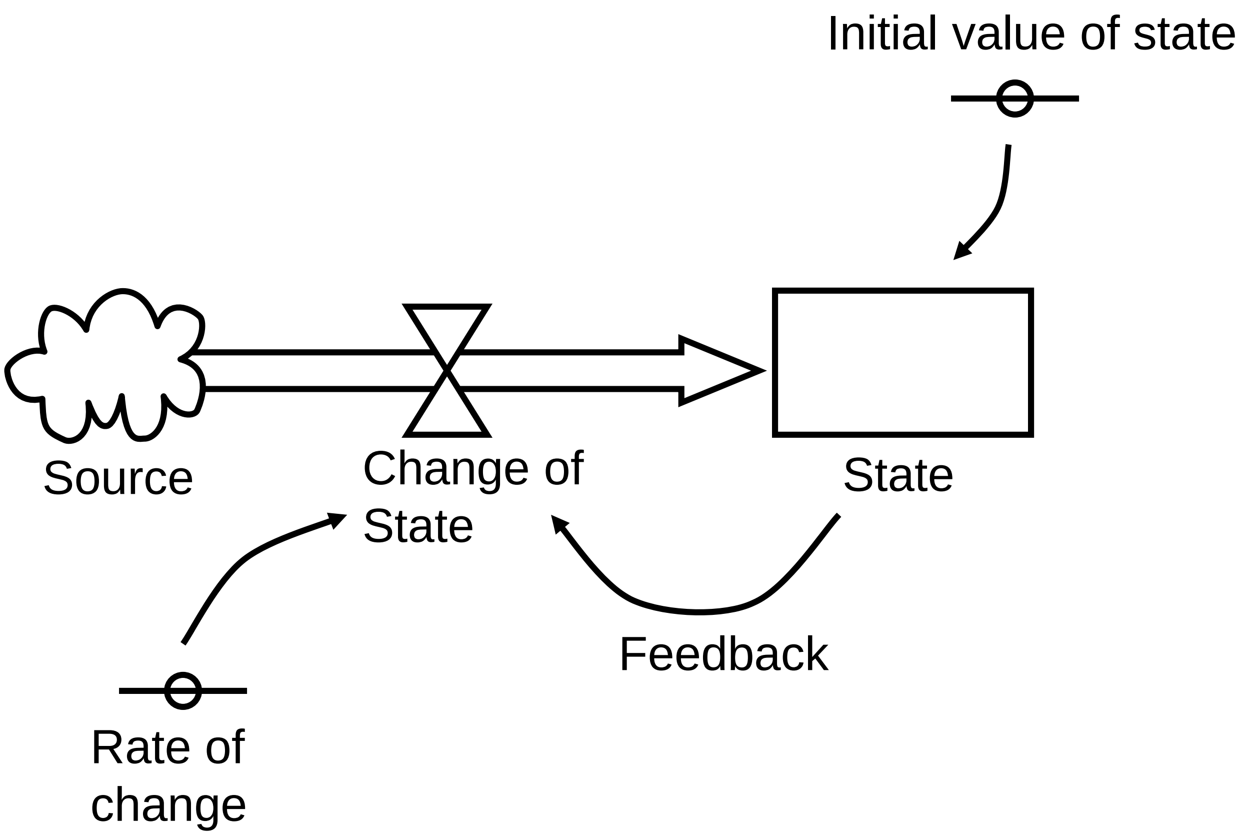
Everything that can directly/analytically be calculated out of stocks and constants. Often useful, to avoid confusing models.

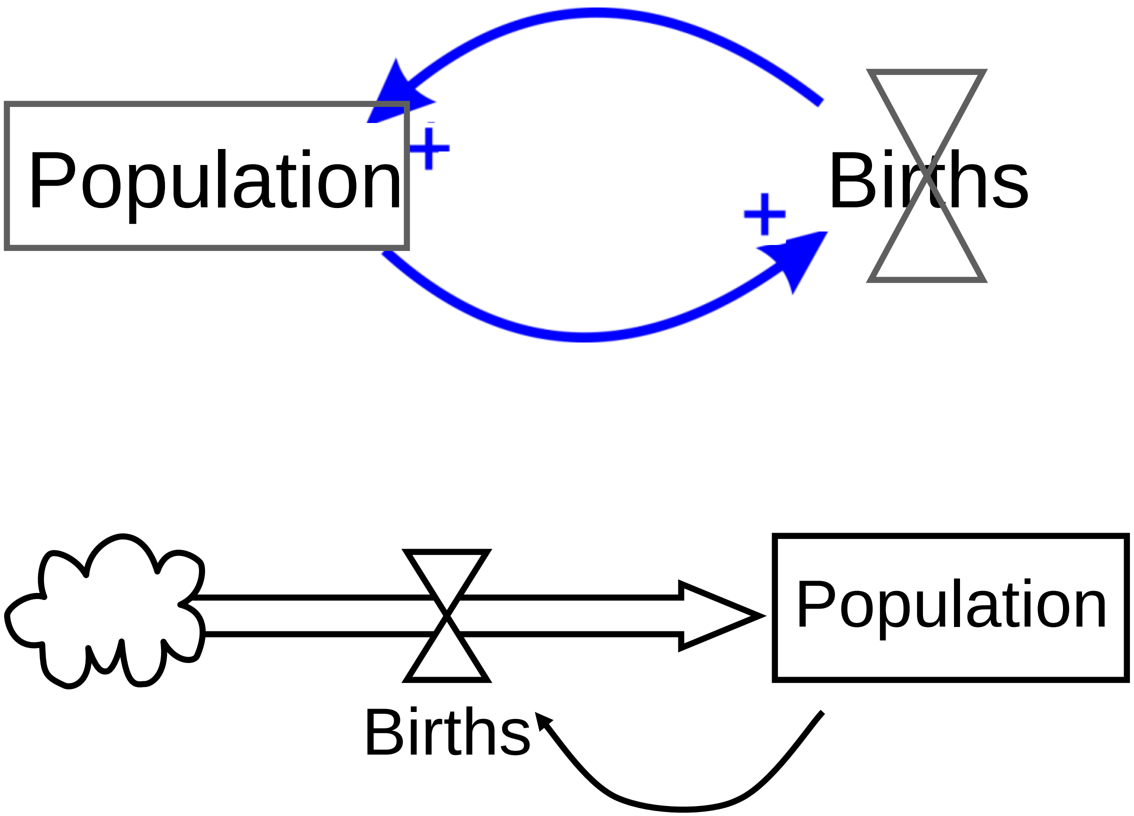
E.g.: Density (can directly be calculated by the stocks/constants „mass“ and „volume“), Quelength (calculated by stock „people in queue“ and constant „average size of one person“), etc.

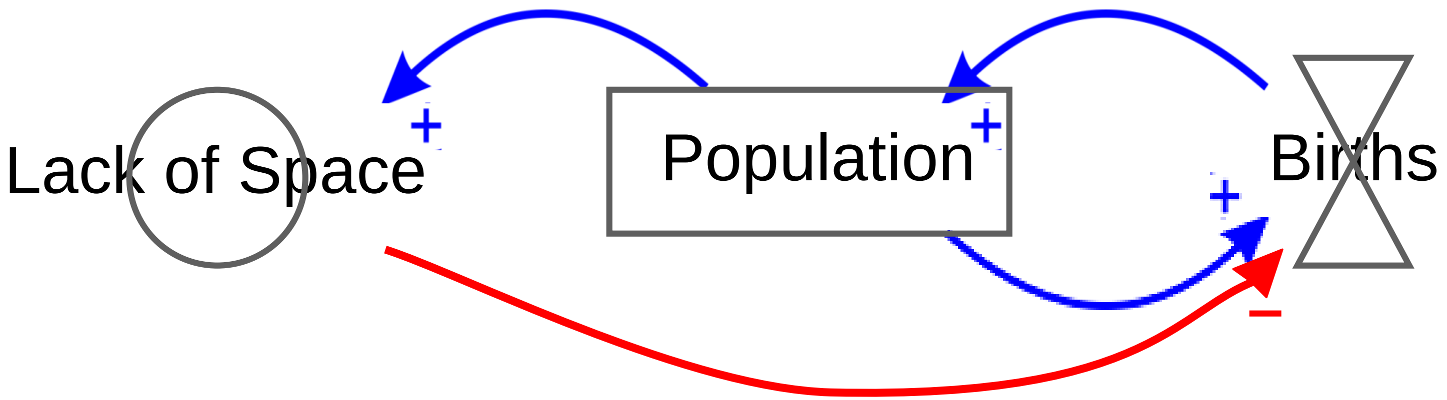
Parameter /Constant

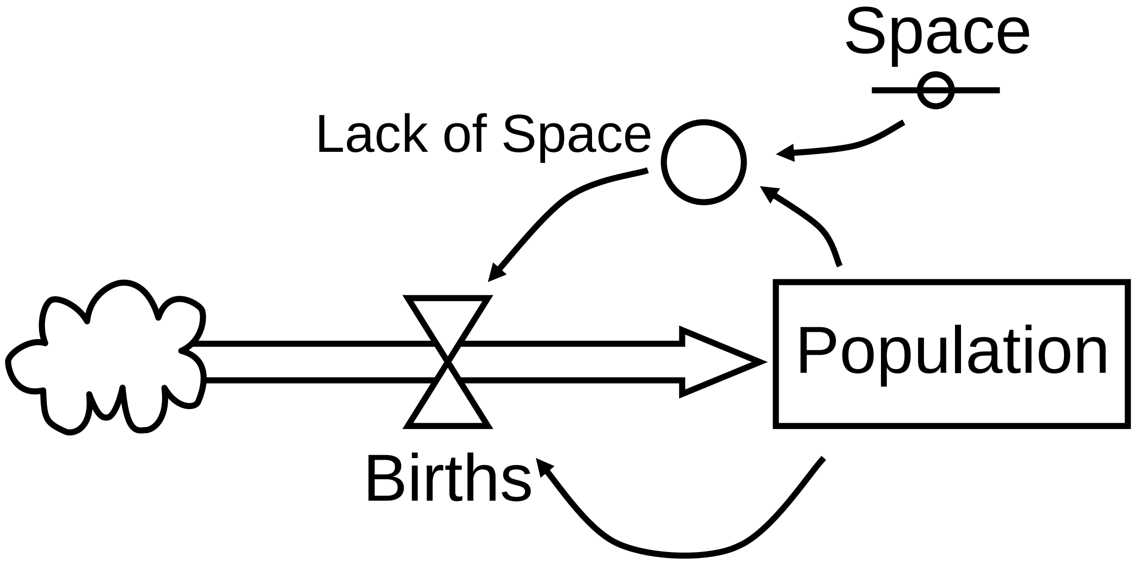
Everything that is predefined for the whole simulation – usually it is a constant but can be a function too.

E.g.: Average Temperature, Number of Cash Desks (In a supermarket), Birth Rate, Maximum capacity of a Room, etc.

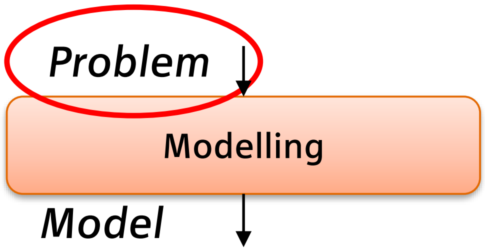

Initial value of state
Source
Change of
State
State
Feedback
Rate of
change

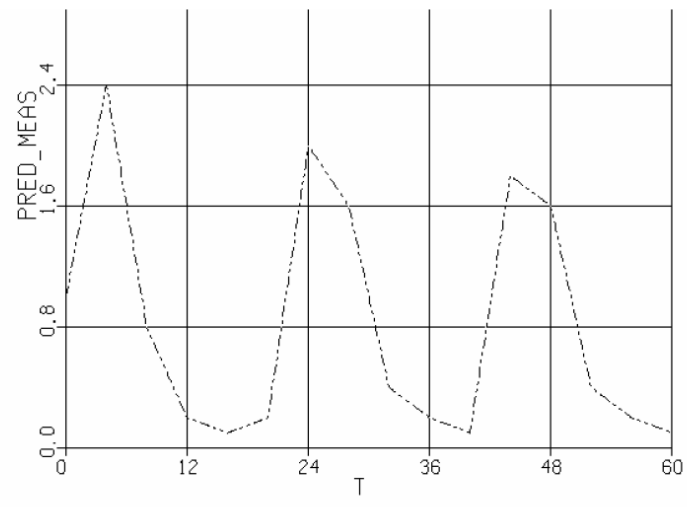

Population
+
Births
+
Population
Births


Lack of Space
+
Population
+
Births
+
–


Space
Lack of Space
Population
Births

Predator – Prey System





Dynamics: Predator eats Prey
Predator / Prey births, deaths

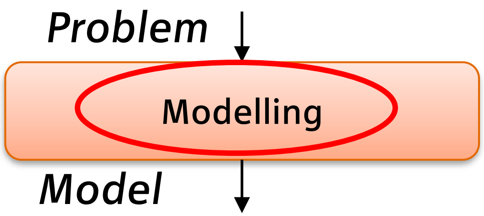
Environment: isolated

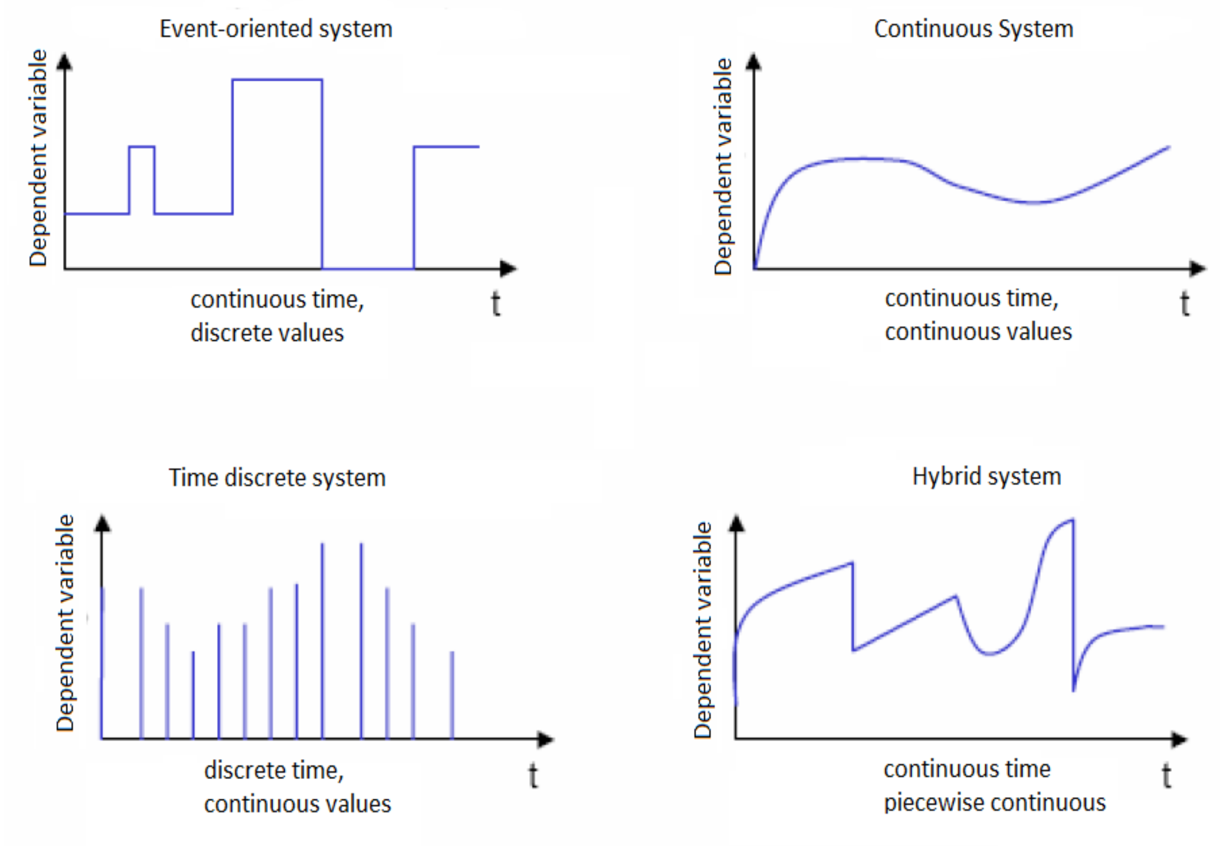
Measurement: Predator Population

5 Years = 60 months, quarterly

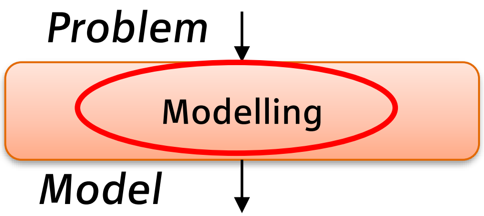
Problem: When is a reasonable time to use chemical pesticides to reduce number of predators?

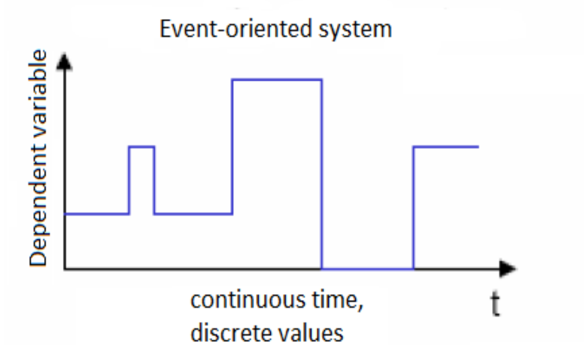
Predator – Prey System

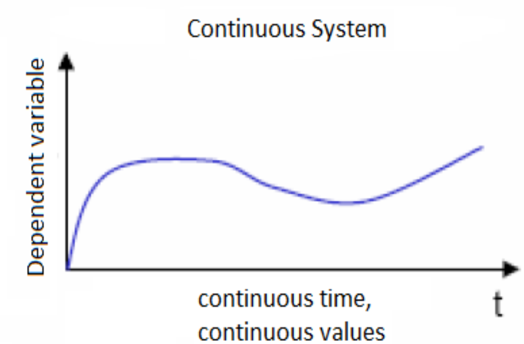




Predator – Prey System

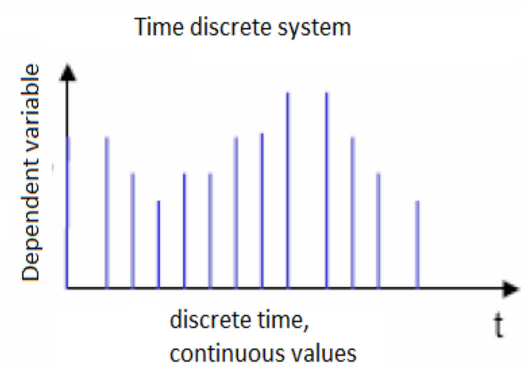


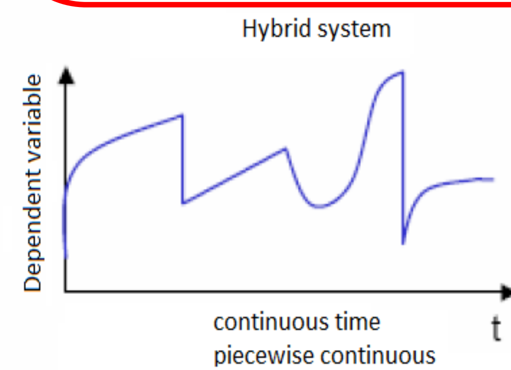




$Y(t)$ ... Prey

$X(t)$ ... Predators

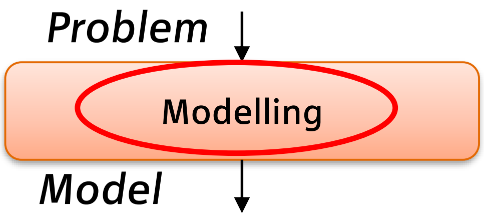


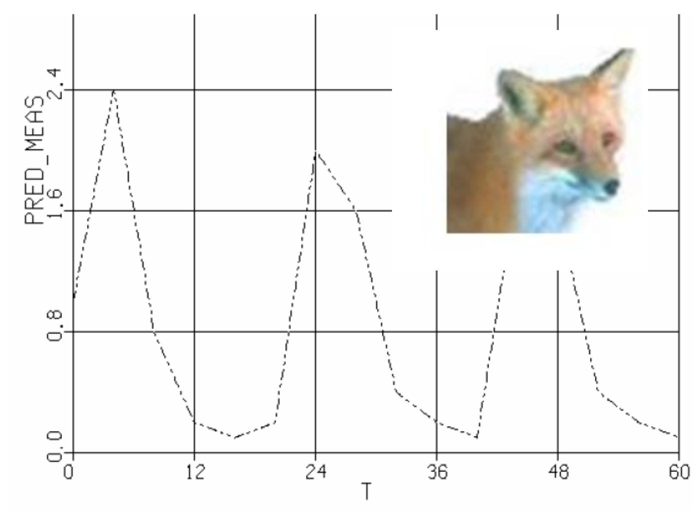


Separation –
Isolated environment

Choice -
2 variables = 2 states


Modelling





Separation –

Isolated environment

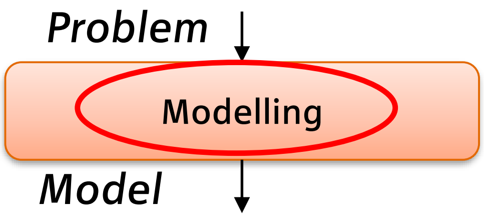
Choice -

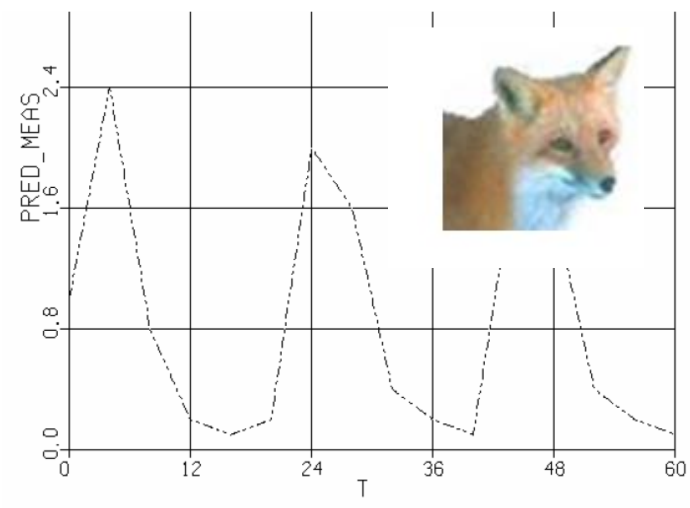
2 variables = 2 states

Causality –

Predator – Prey – Model

Modelling





Separation –

Isolated environment

Choice -

2 variables = 2 states

Causality –

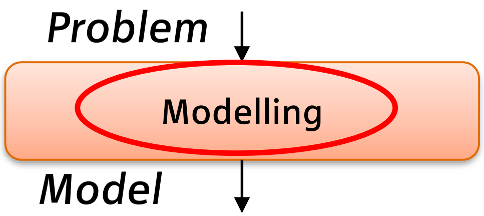
Predator – Prey – Model

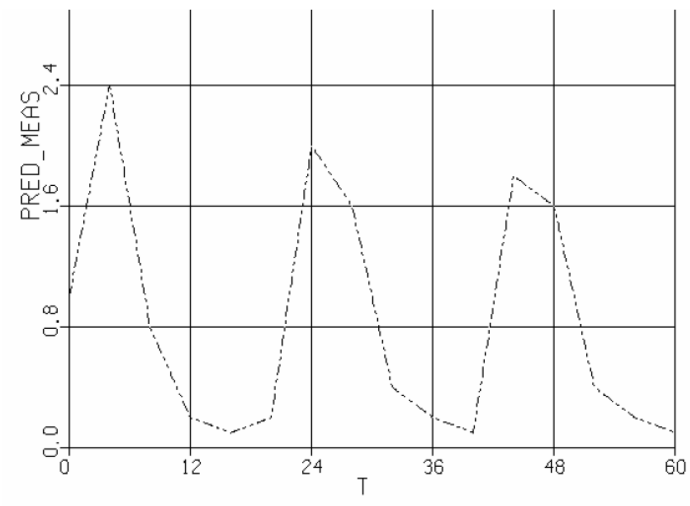
Y(t) .. Prey Population

X(t) .. Predator Population

Modelling

Mathematical Modelling and Simulation





Causality –

Predator – Prey – Model

Y(t) .. Prey Population

X(t) .. Predator Population

System Dynamics –

Population Interaction

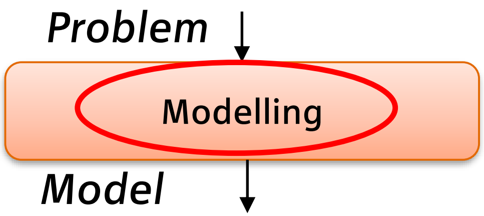
Modelling

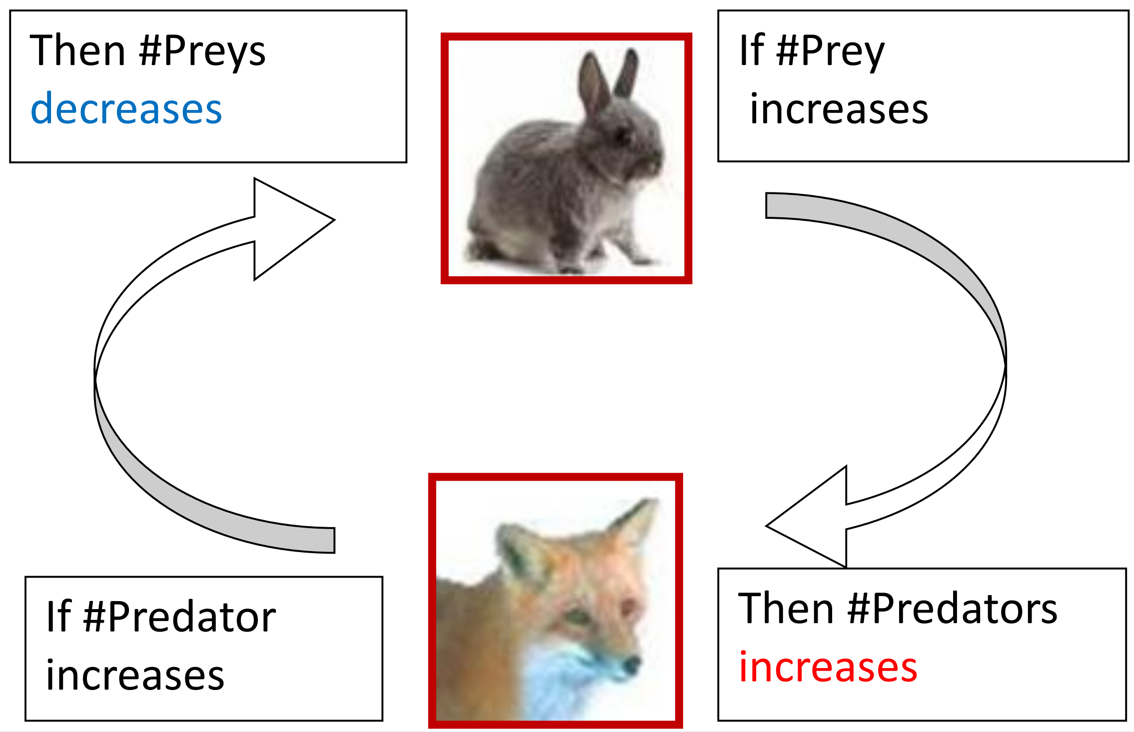
Causality – Predator – Prey – Model

Y(t) .. Prey,

X(t) .. Predator

System Dynamics - Population interaction





Modelling

Mathematical Modelling and Simulation

Problem ↓

Modelling

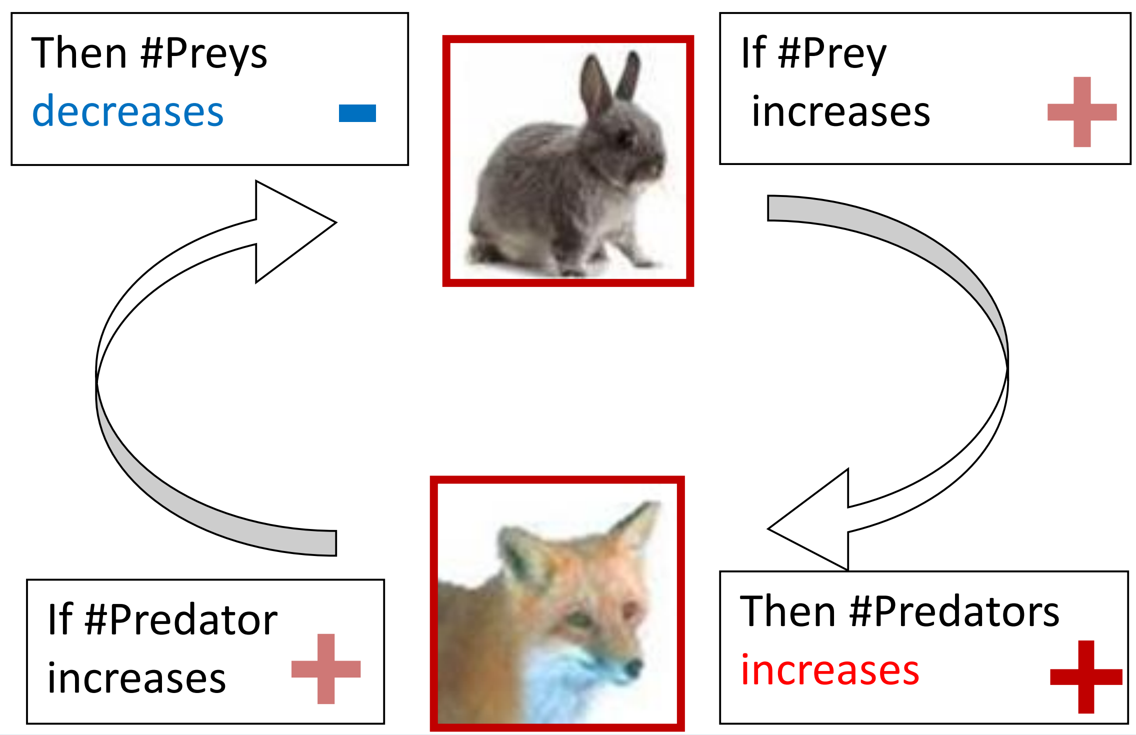
Model ↓

Causality – Predator – Prey – Model

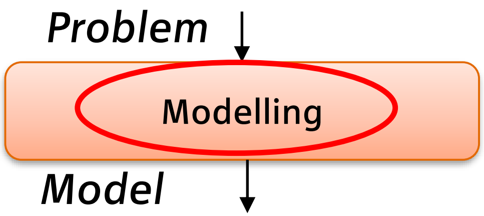
Y(t) .. Prey,

X(t) .. Predator

System Dynamics - Population interaction

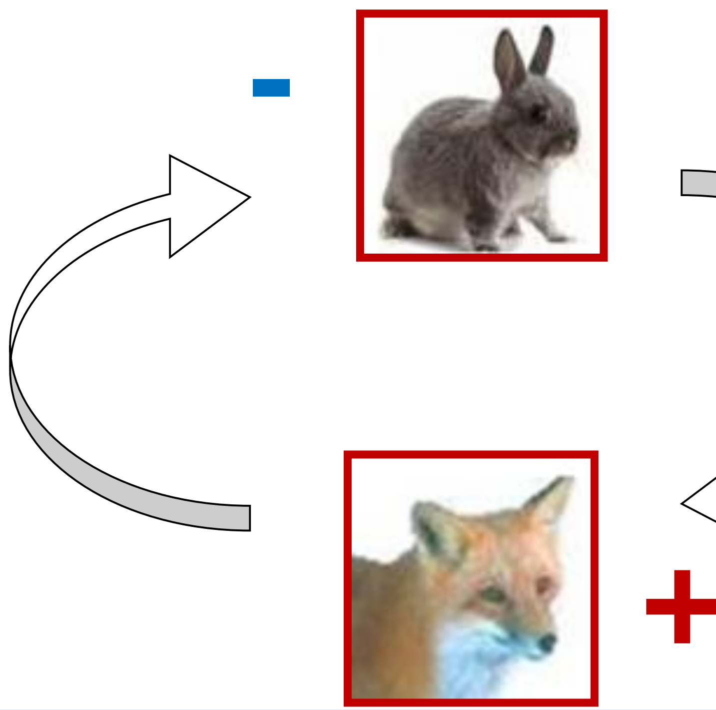


Modelling



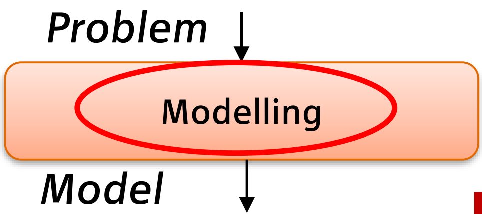
Causality – Predator – Prey – Model
Y(t) .. Prey,
X(t) .. Predator

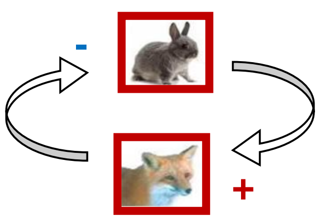
System Dynamics - Population interaction

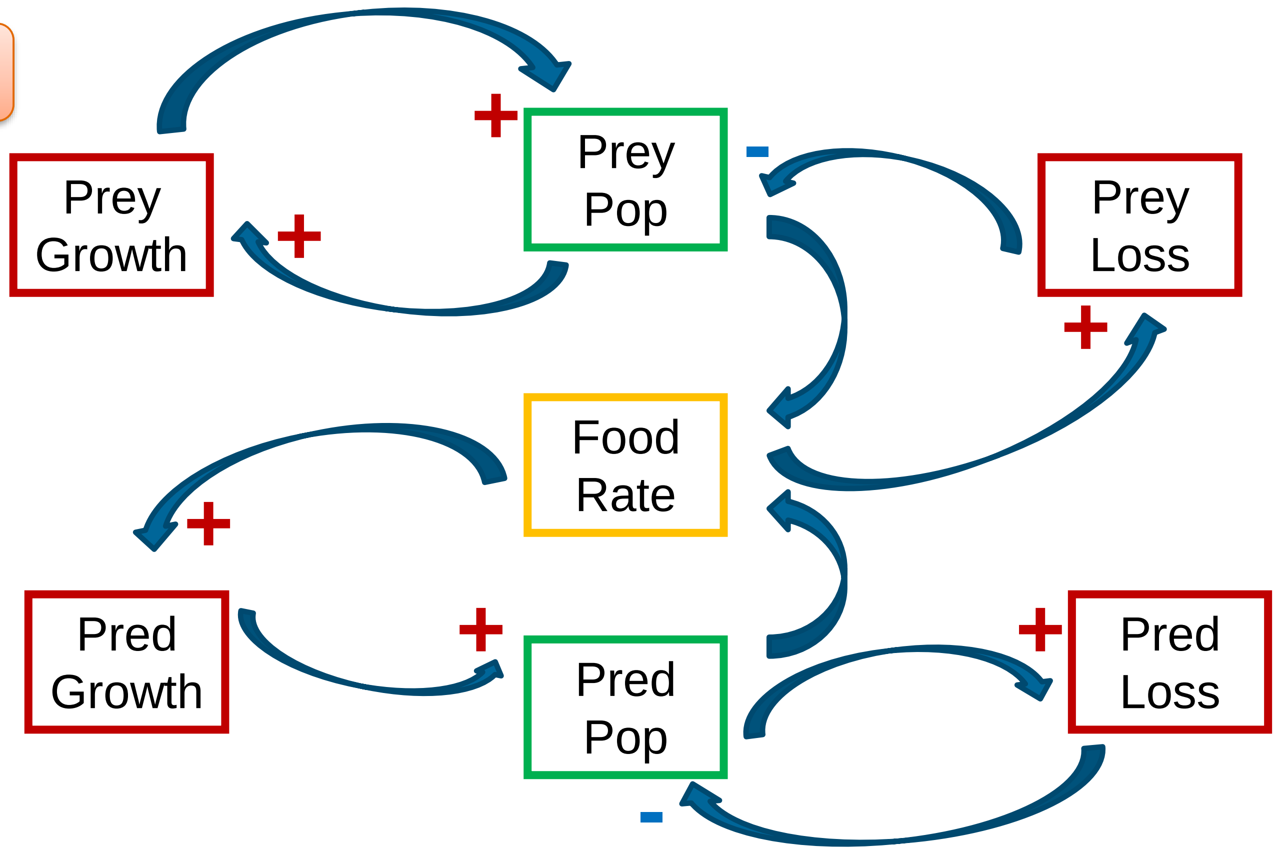


Causal Loop Diagram

System Dynamics - Population interaction



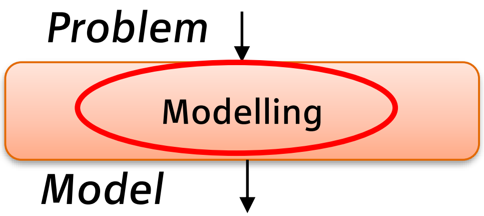


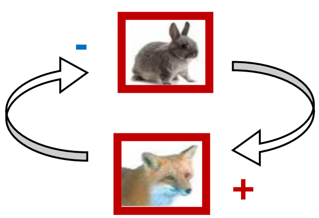


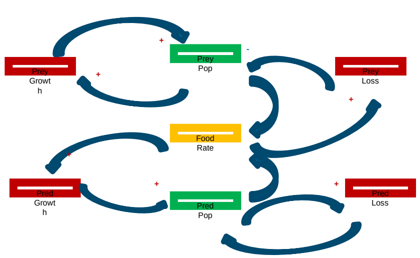
Causal Loop Diagram

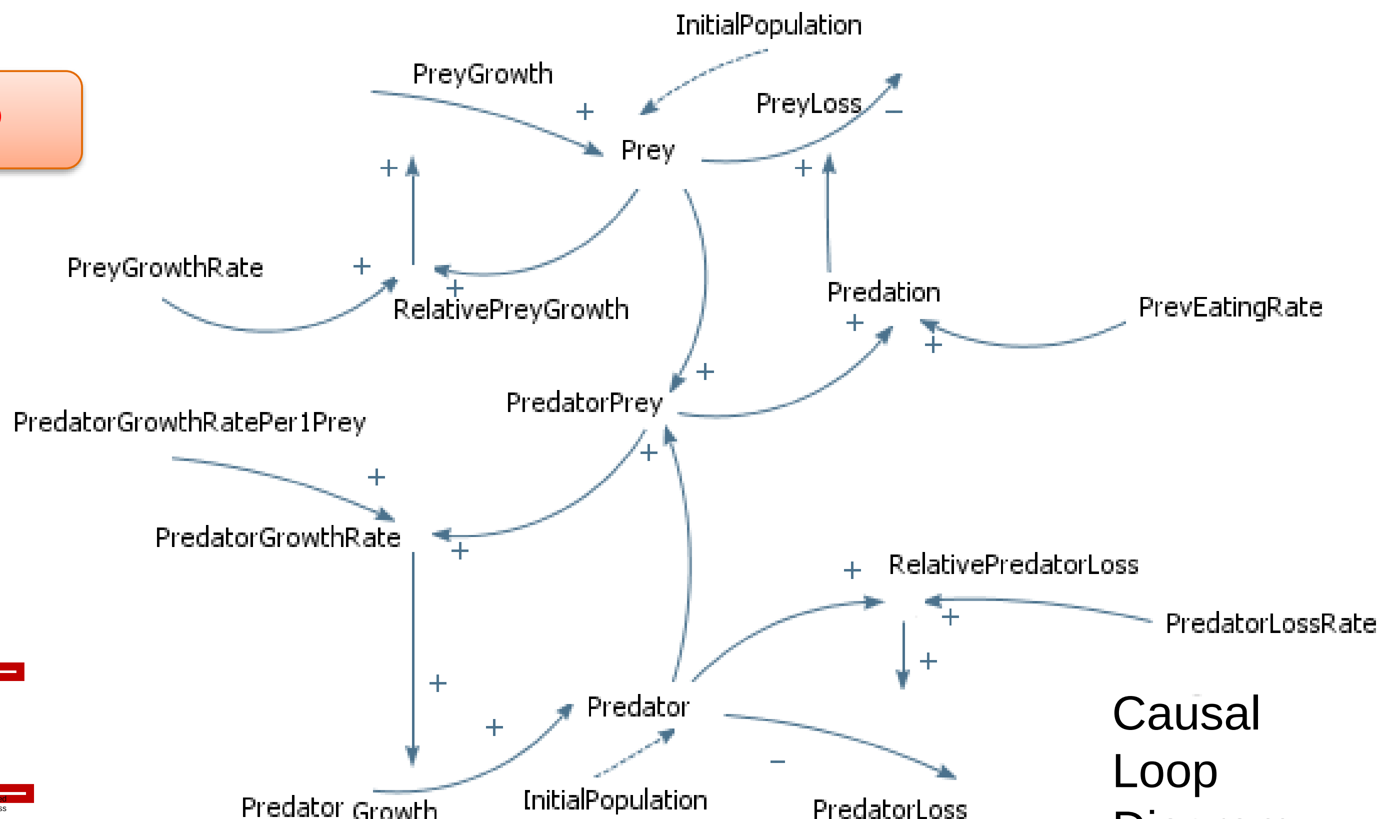
Mathematical Modelling and Simulation

System Dynamics - Population interaction



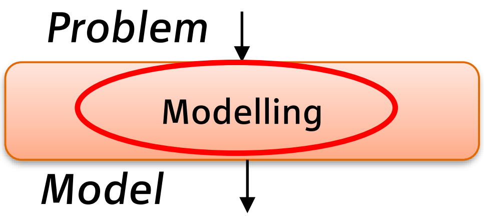


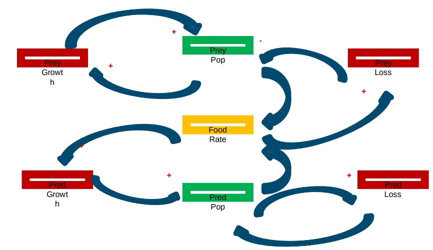


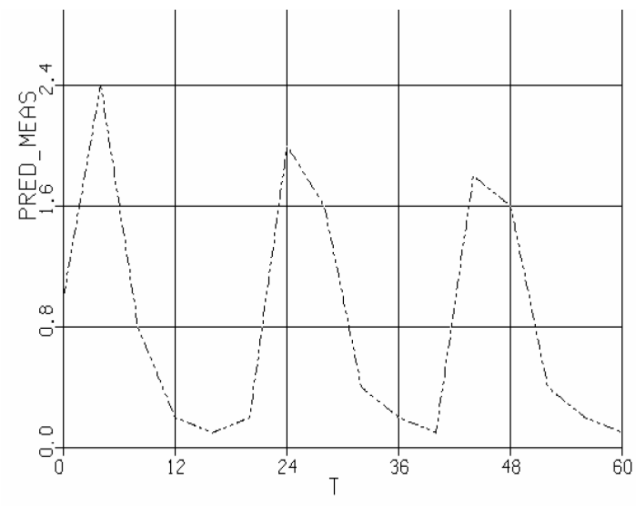


Causal Loop Diagram

Modelling







Causality –

Predator – Prey – Model

x(t) .. Prey

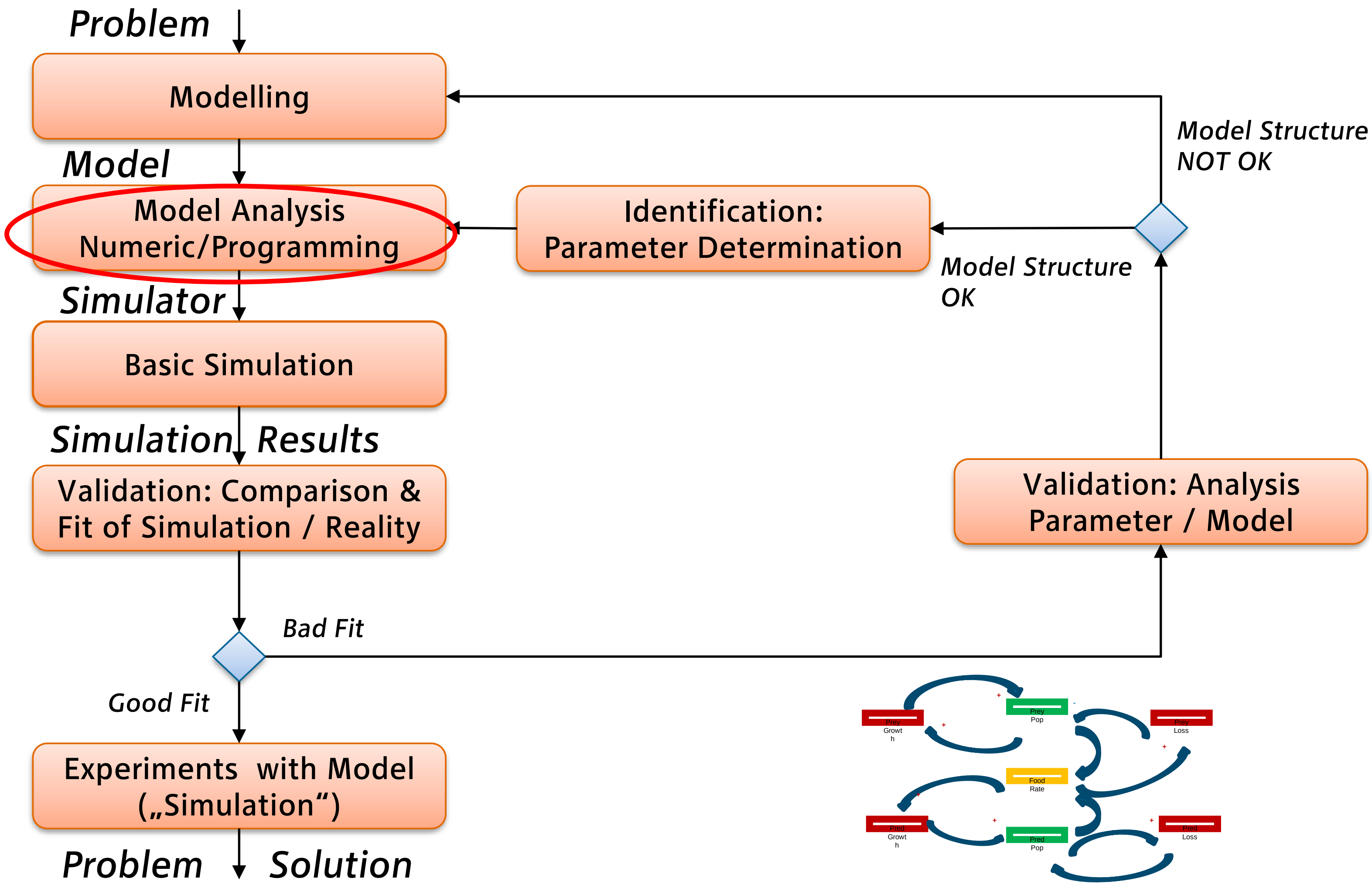
y(t) .. Predator

Logistic Growth -

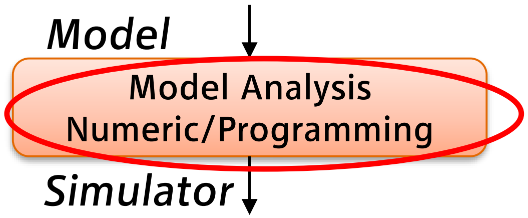
$$\dot{x} = ax - bxy$$
$$\dot{y} = -dy + exy$$

Population rate = Growth rate + food rate

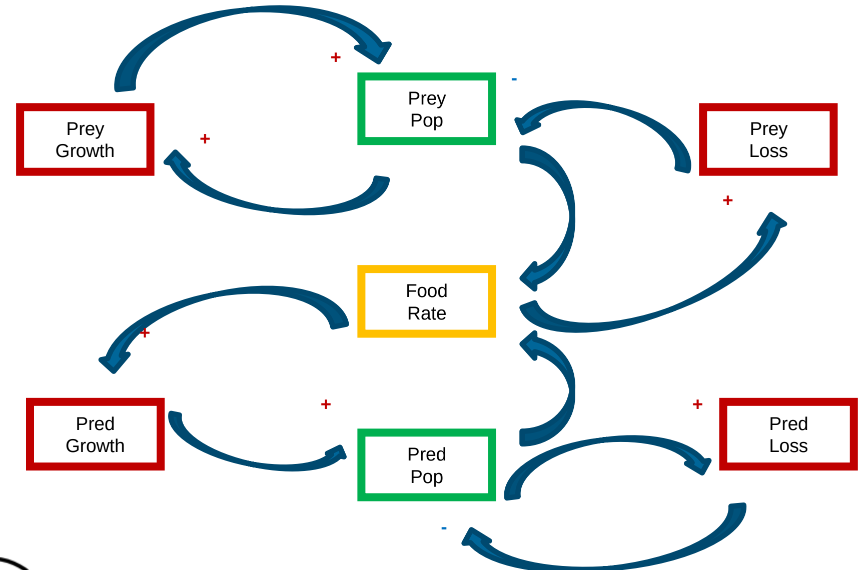
Simulation Circle: Predator - Prey



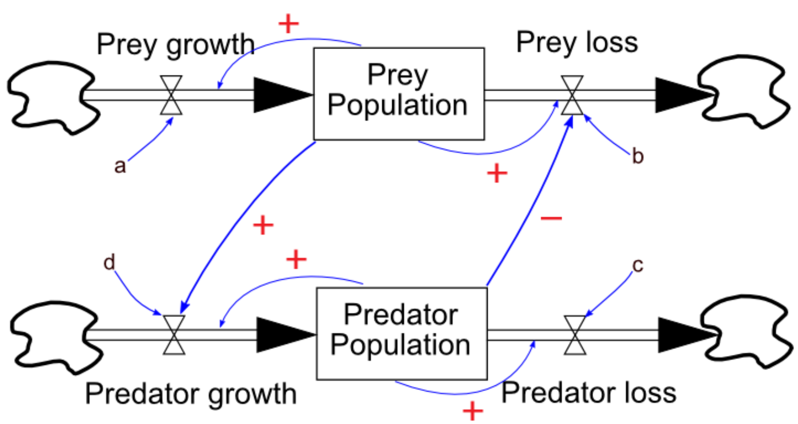
Model Analysis



Causal Loop Diagram

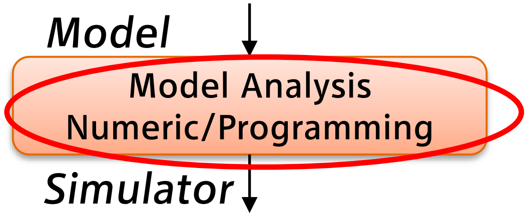


Stock and Flow Diagram

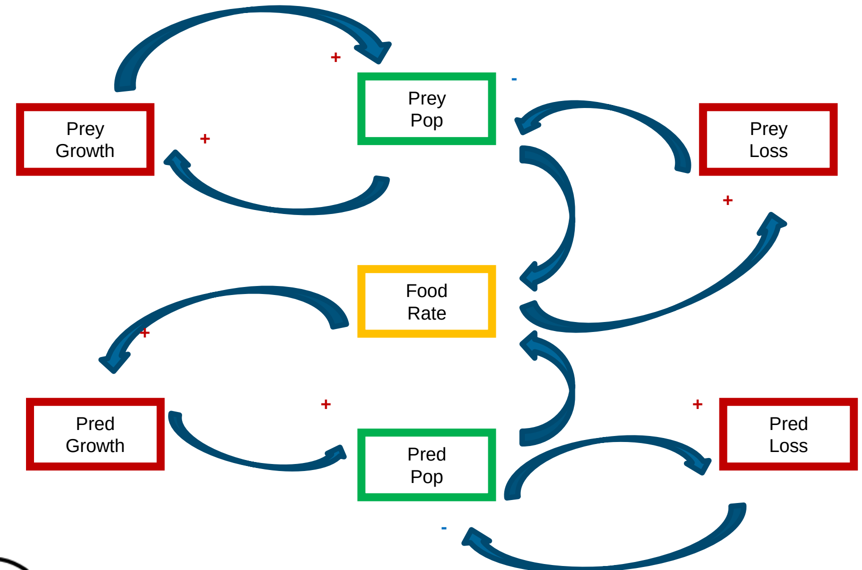


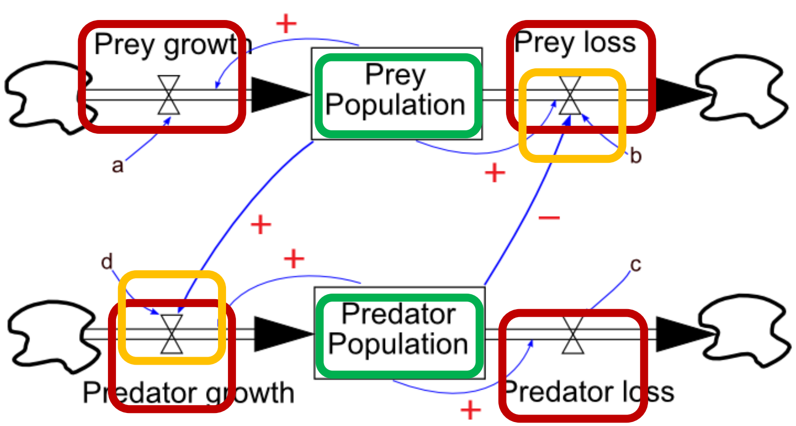
Model Analysis

Mathematical Modelling and Simulation



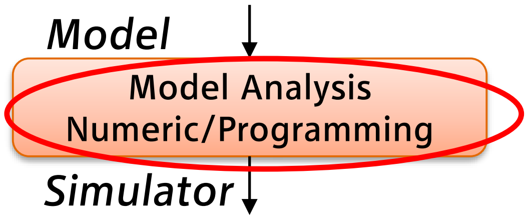
Causal Loop Diagram

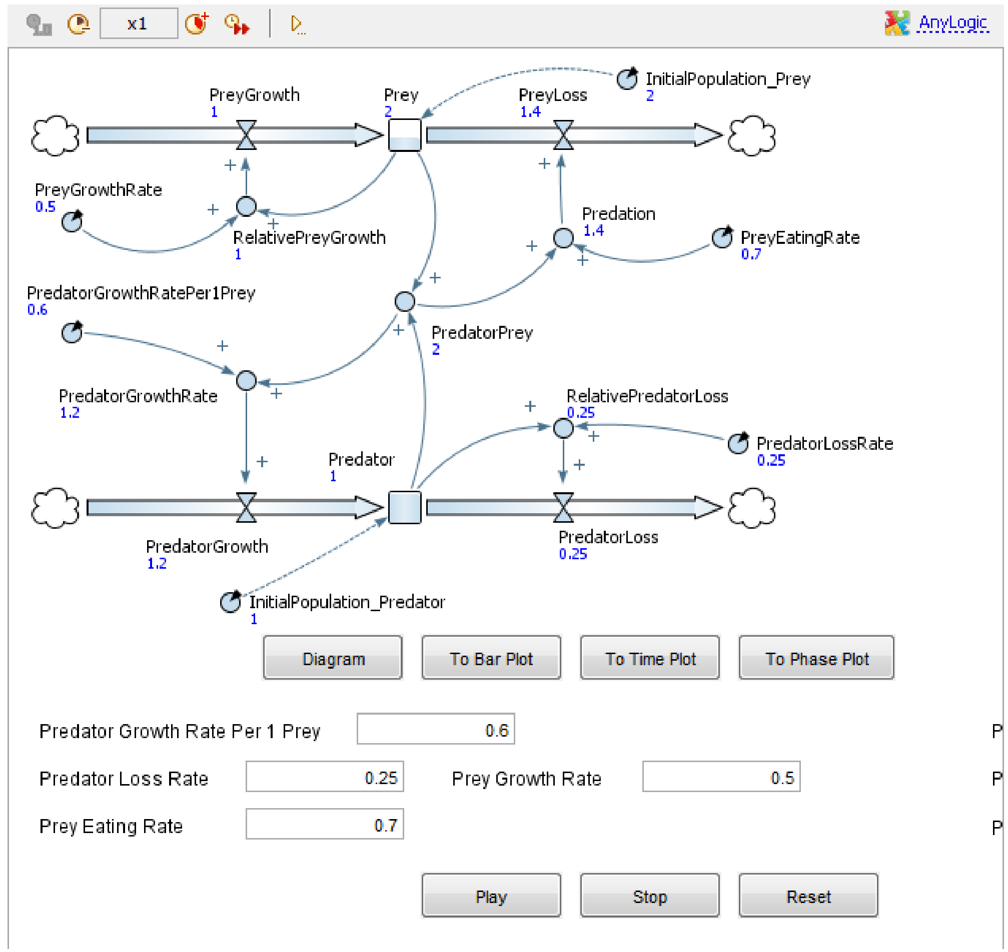




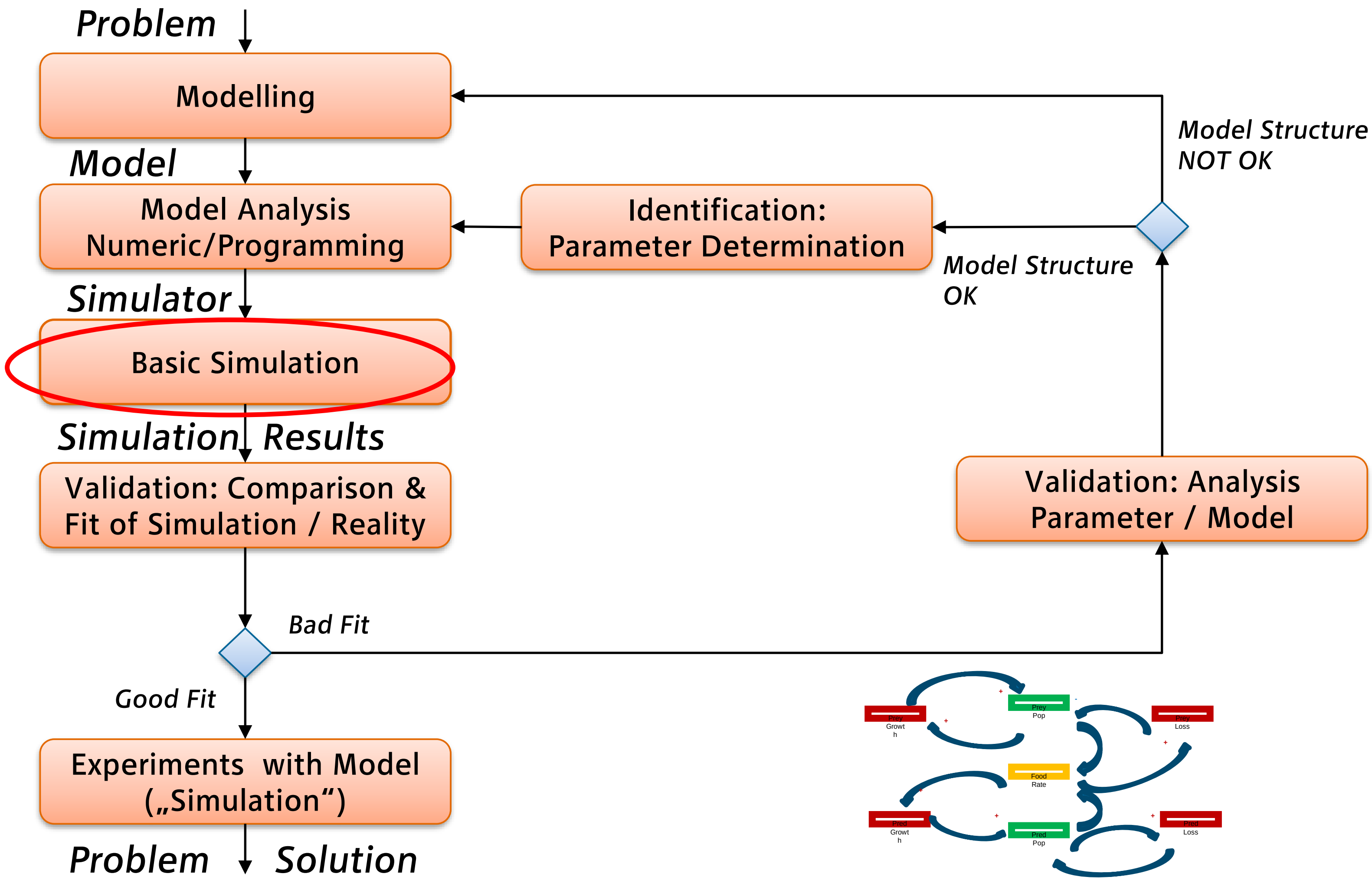
Stock and Flow Diagram

Implementation



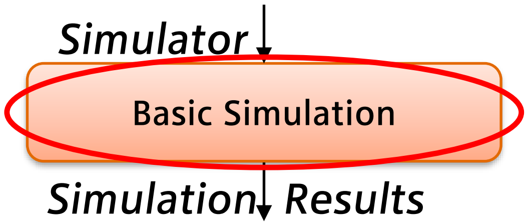


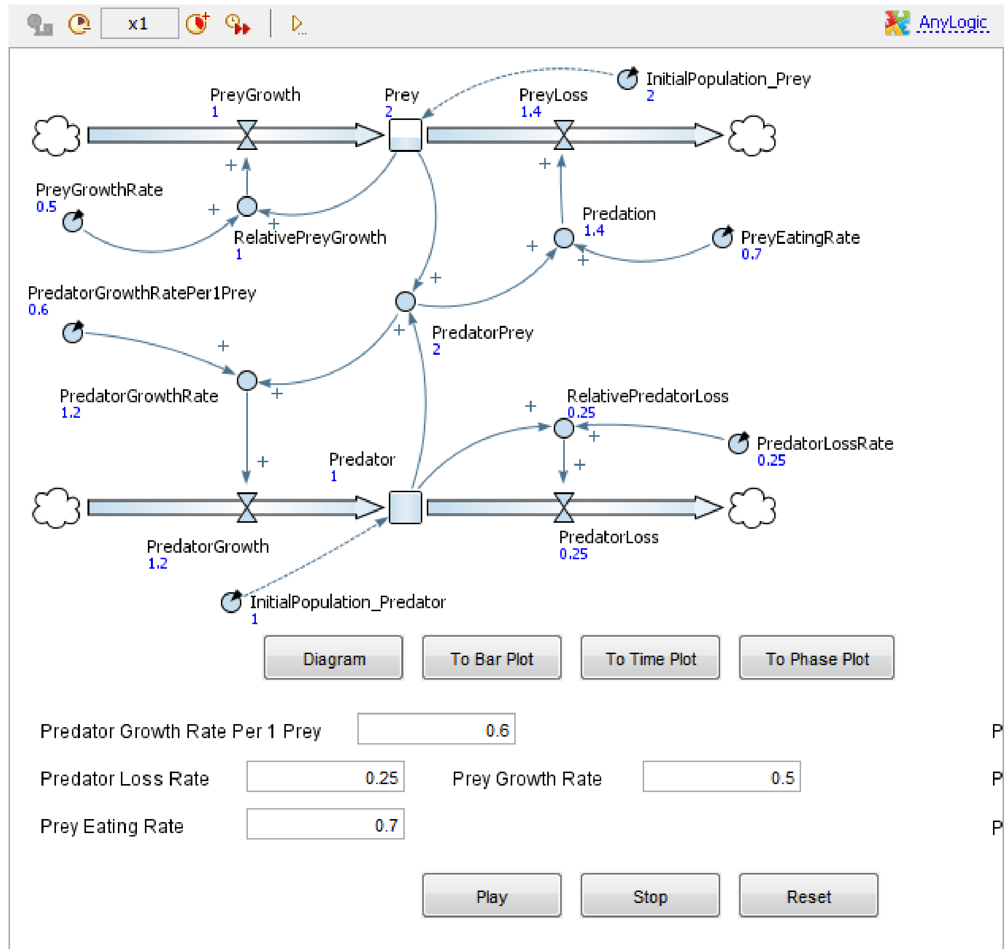
Simulation Circle: Predator - Prey



Implementation

Mathematical Modelling and Simulation





Implementation

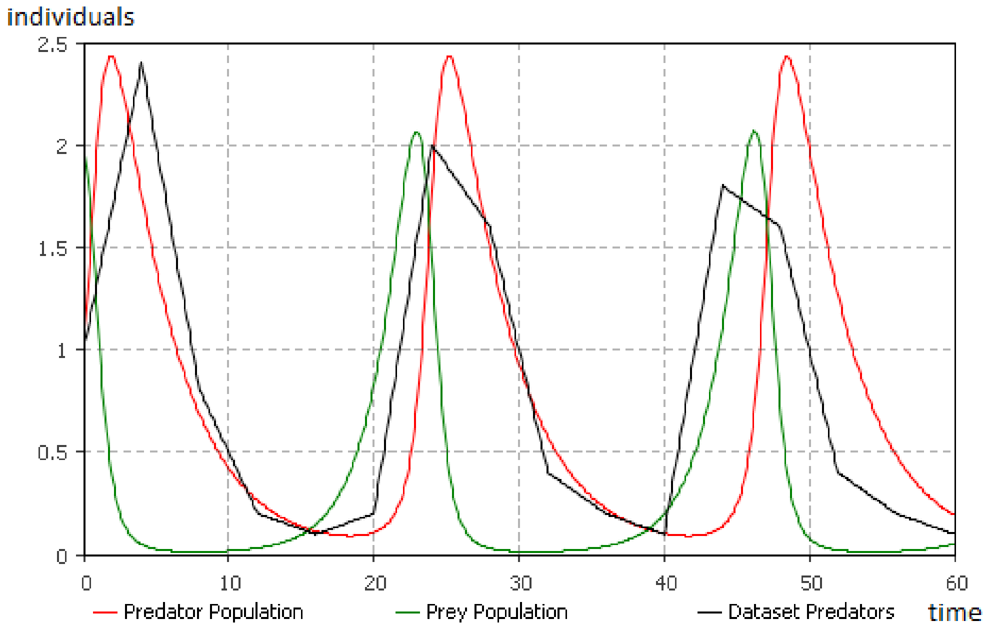
Mathematical Modelling and Simulation

Simulator

Basic Simulation

Simulation Results

Population development over time:



Parameters:

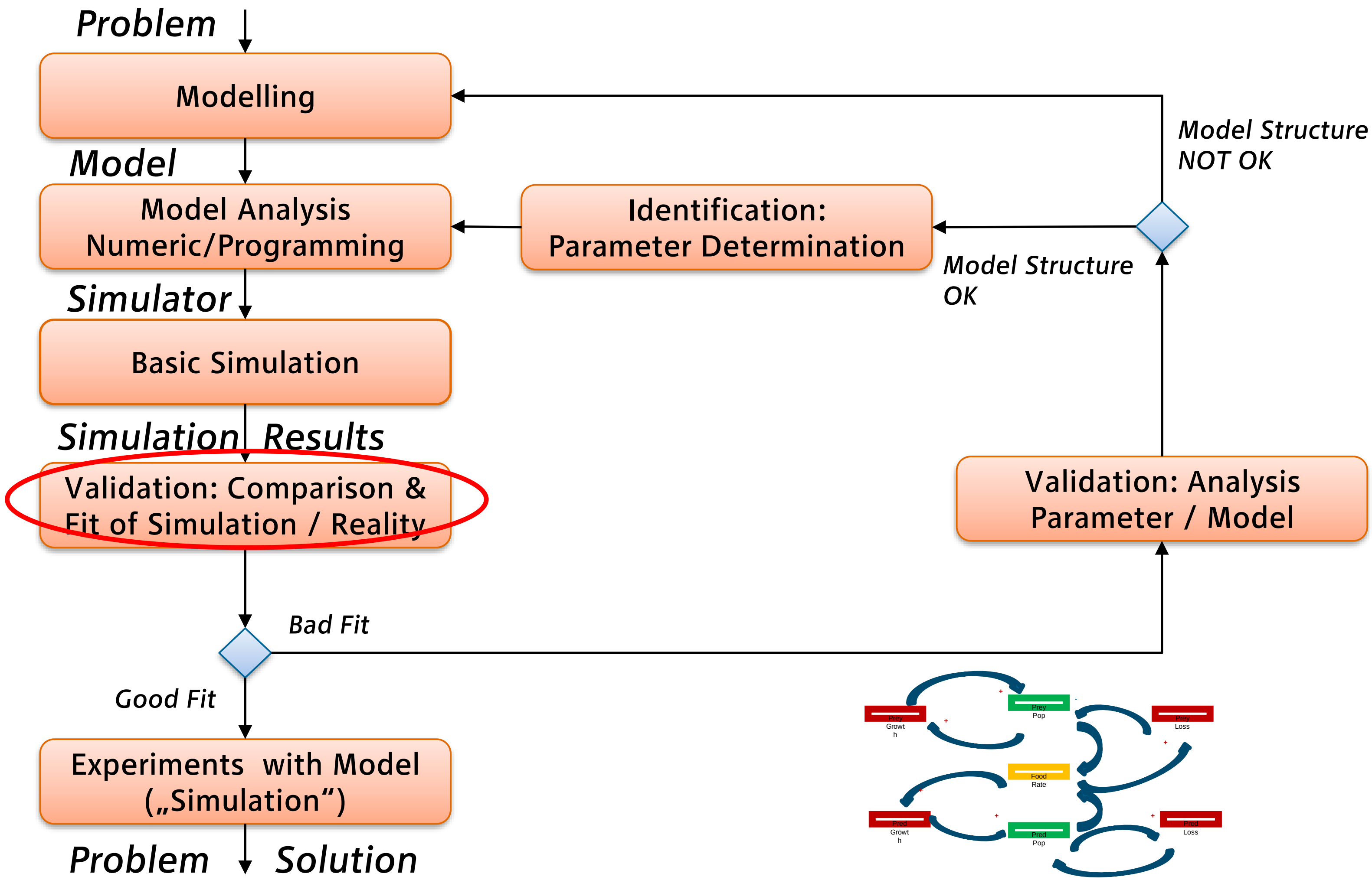
Predator Growth Rate Per 1 Prey	0.6
Predator Loss Rate	0.25
Prey Eating Rate	0.7
Prey Growth Rate	0.5

$$\dot{x} = (a - b \cdot y)x$$

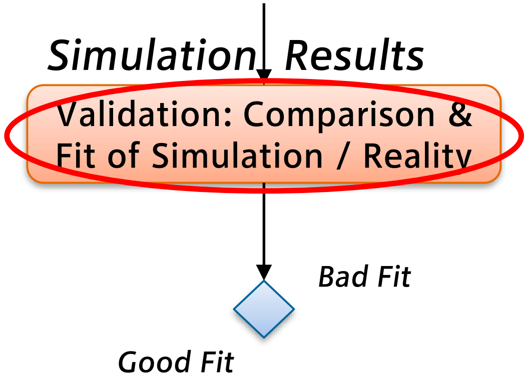
$$\dot{y} = (-c + d \cdot x)y$$

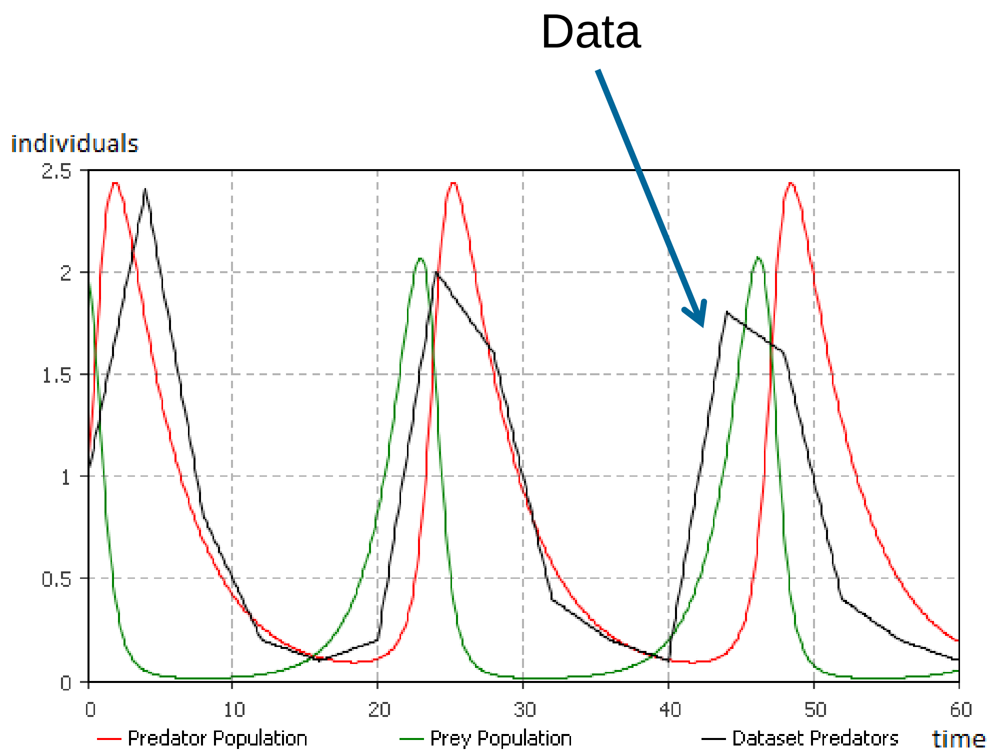
Simulation Circle: Predator - Prey

Mathematical Modelling and Simulation

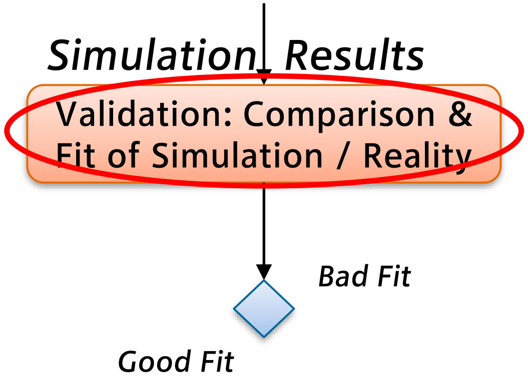


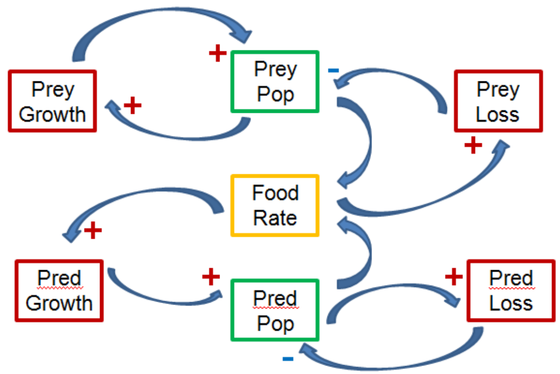
Validation

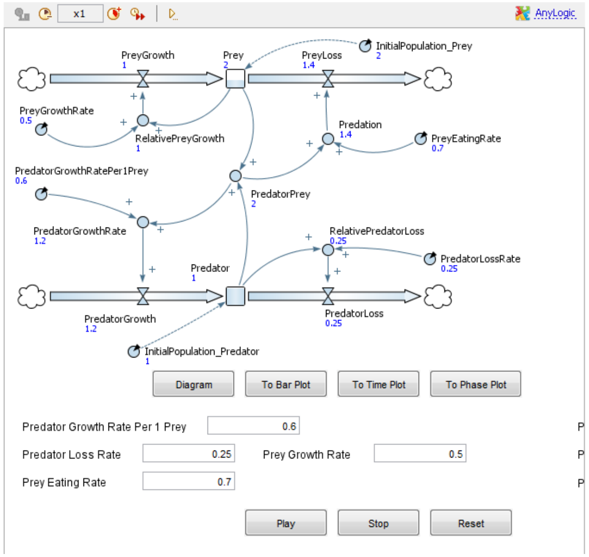

Simulation Results
Validation: Comparison & Fit of Simulation / Reality
Bad Fit
Good Fit

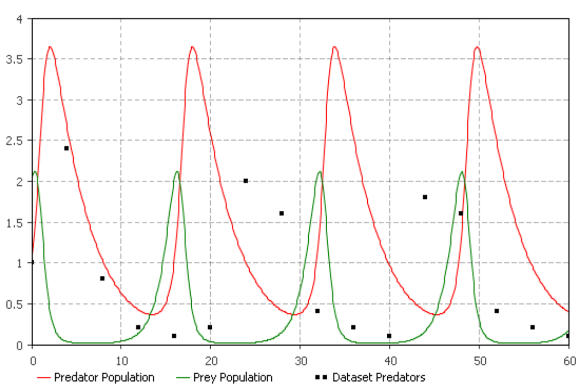

Data
individuals
2.5
2
1.5
1
0.5
0
0
10
20
30
40
50
60
time
Predator Population
Prey Population
Dataset Predators

Data & Simulation Results

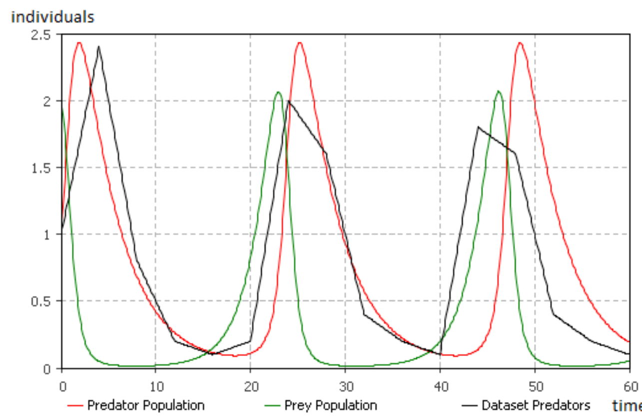






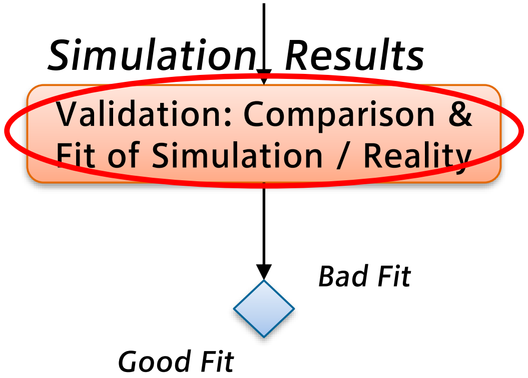


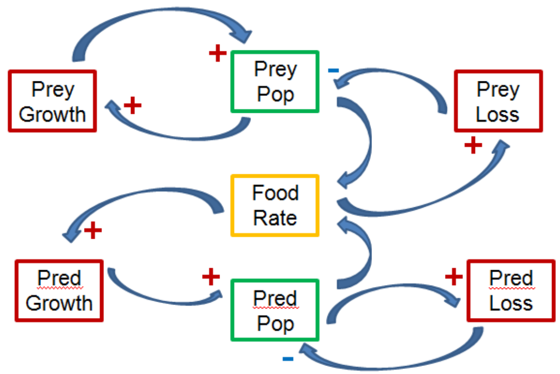
Search for convenient parameters

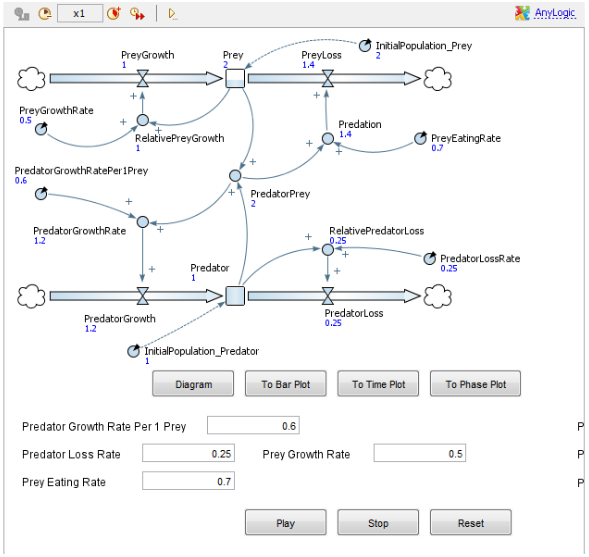


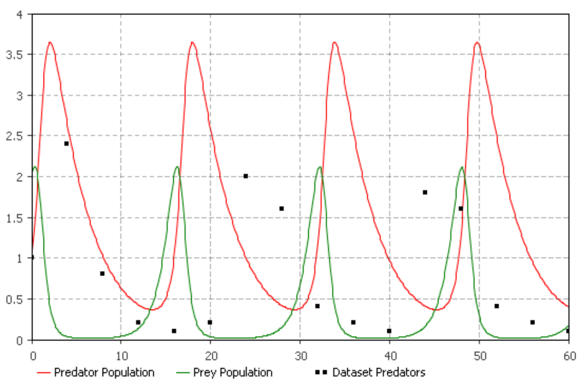
Data & Simulation Results

Mathematical Modelling and Simulation



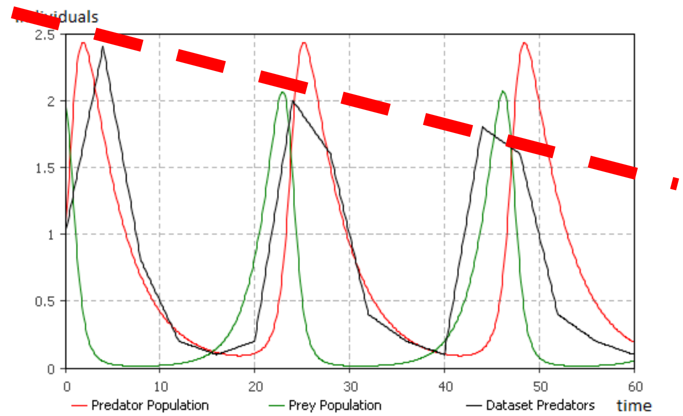




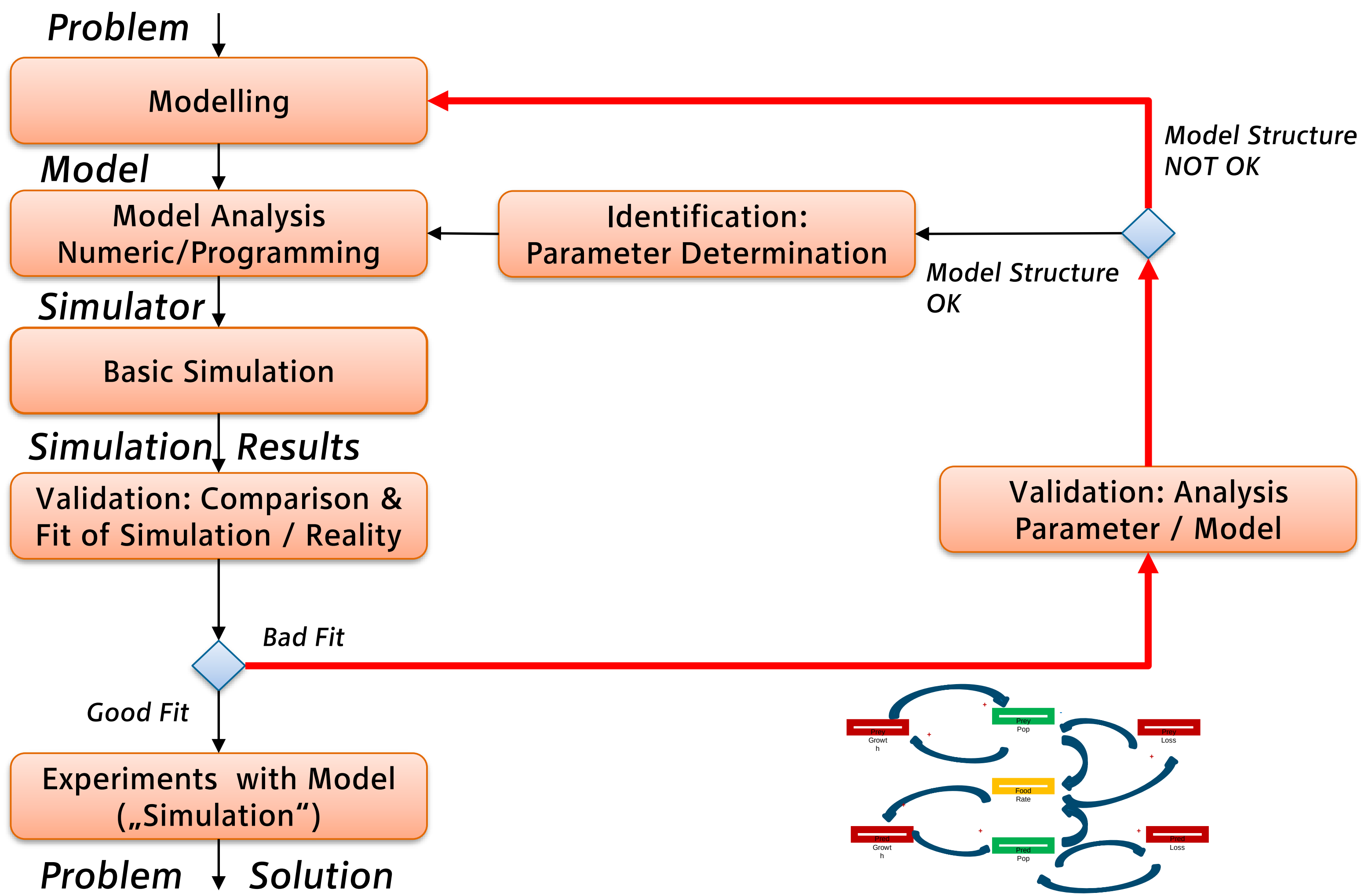


Search for convenient parameters

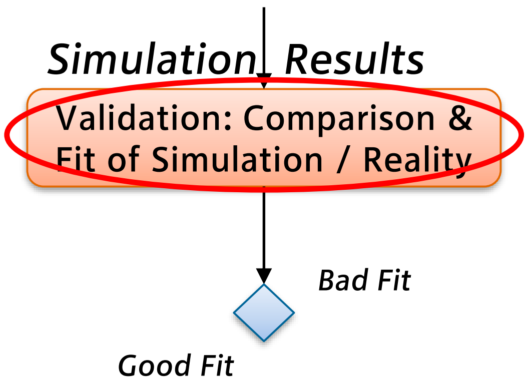
No Damping in Model

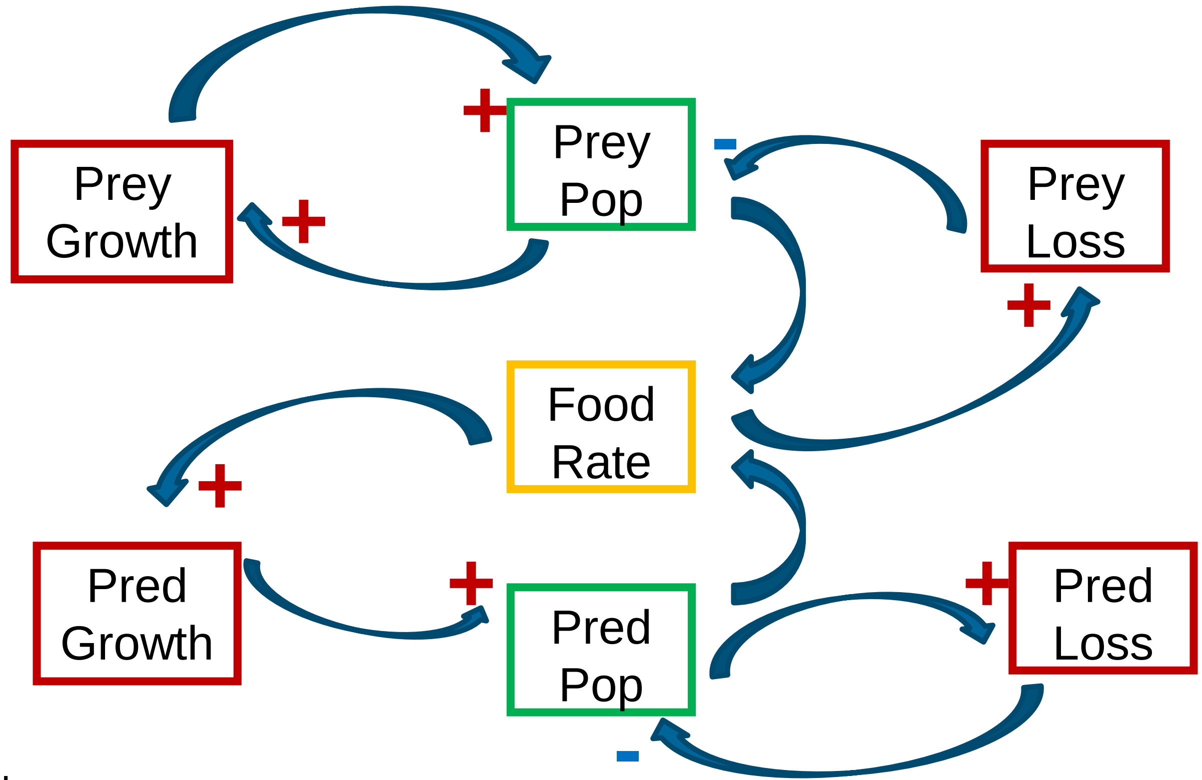


Simulation Circle: Predator - Prey



Model Modification



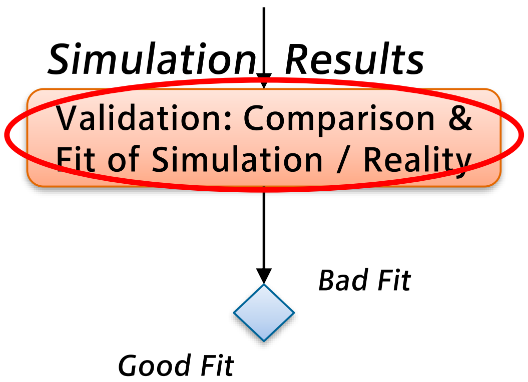


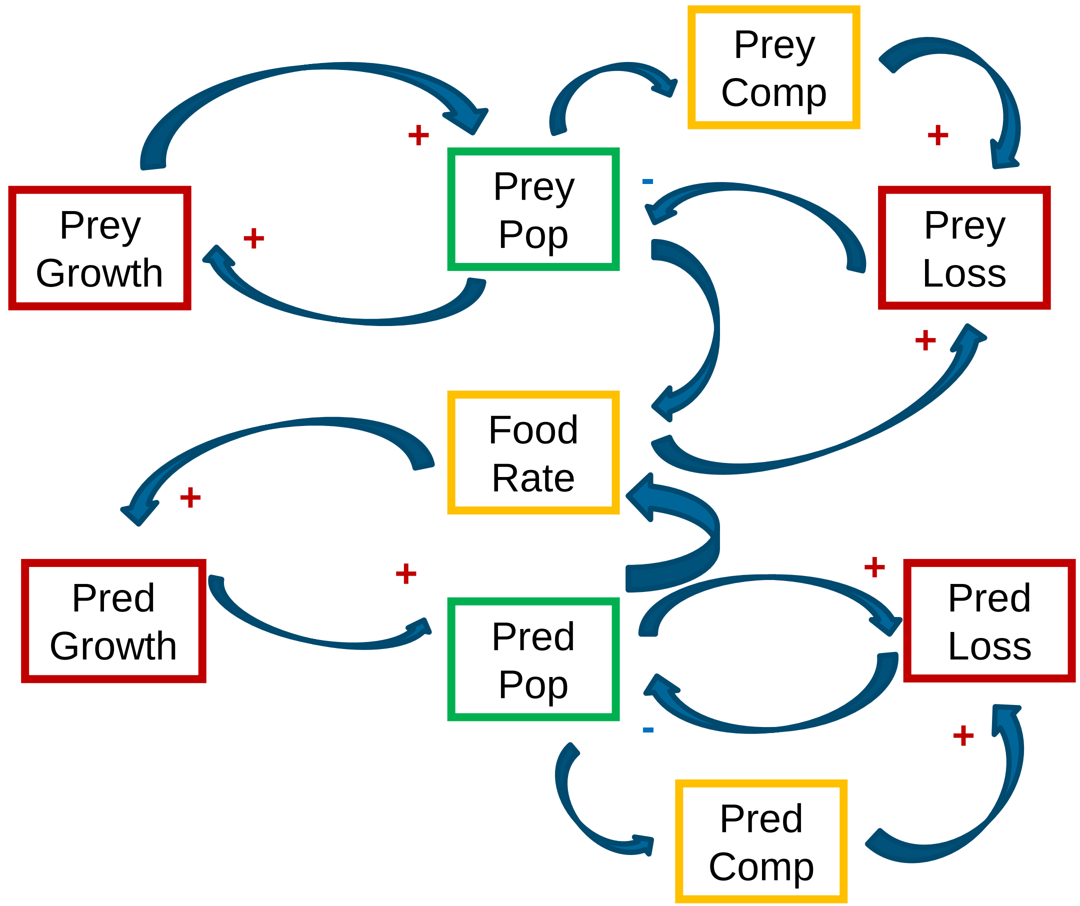
Model Extension:

- Both the predator and the prey compete for food and shelter in the forest.

- Competition sets in and the population of each species tends to control itself via a negative effect, that is the population decreases with a rate directly proportional to the present population of that species.

Model Modification





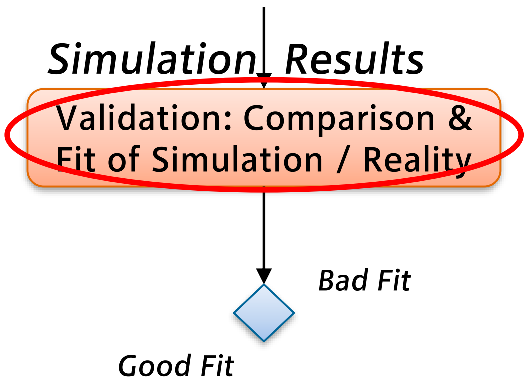
Model Extension:

- Both the predator and the prey compete for food and shelter in the forest.

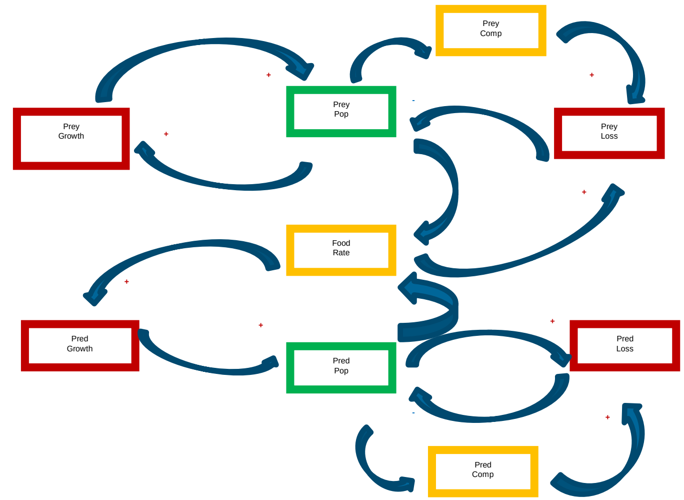
- Competition sets in and the population of each species tends to control itself via a negative effect, that is the population decreases with a rate directly proportional to the present population of that species.

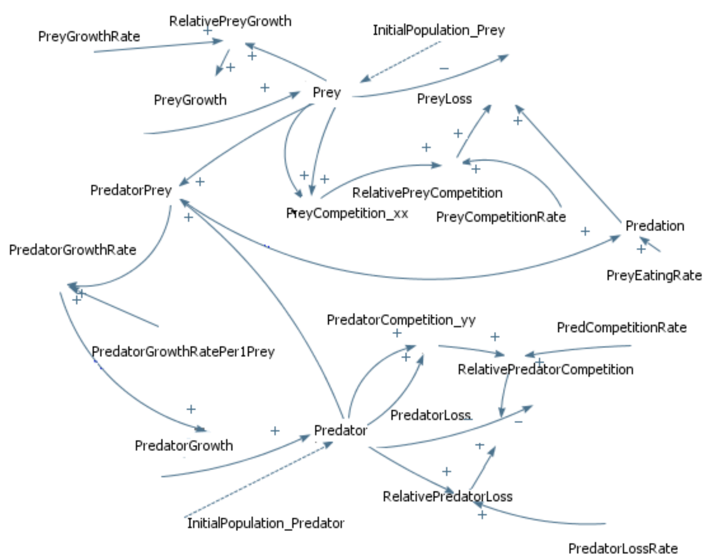
Model Modification

Mathematical Modelling and Simulation



Causal Loop Diagram



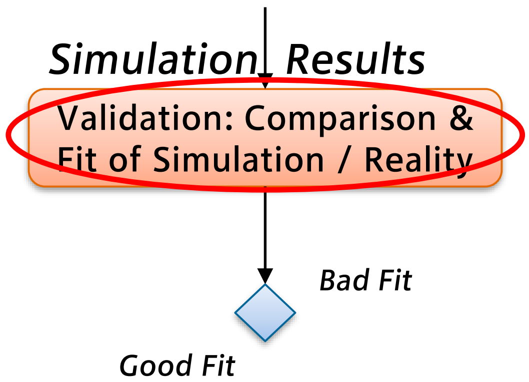


Model Extension:

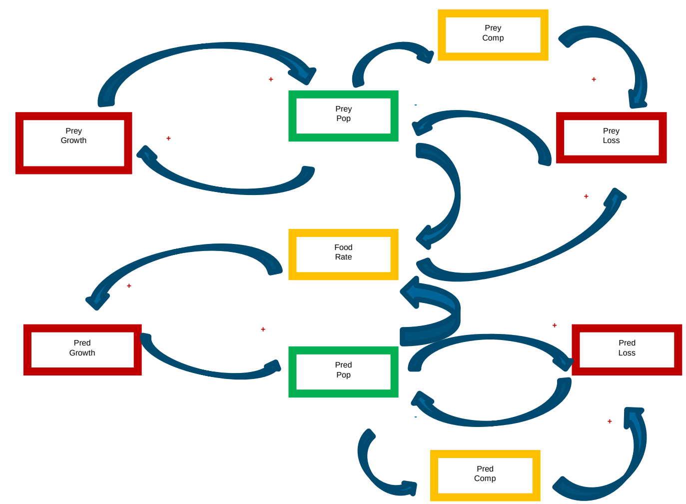
- Competition Feedback

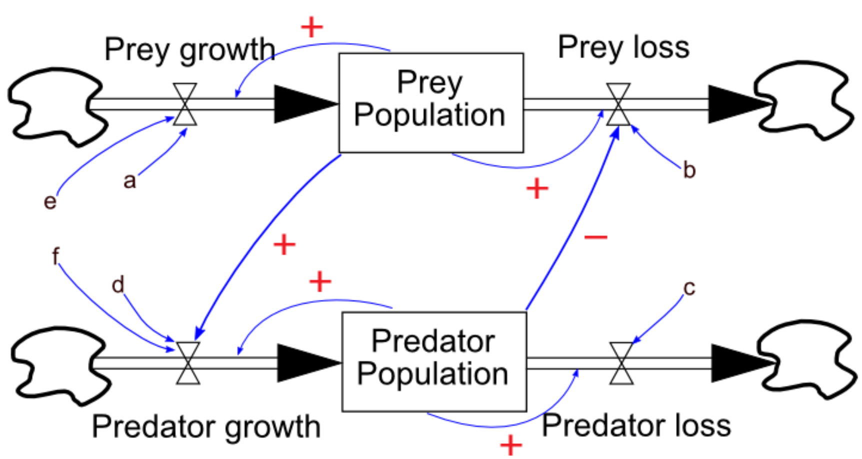
Model Modification

Mathematical Modelling and Simulation



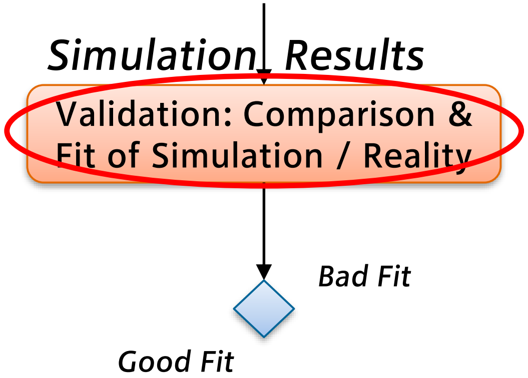
Causal Loop Diagram



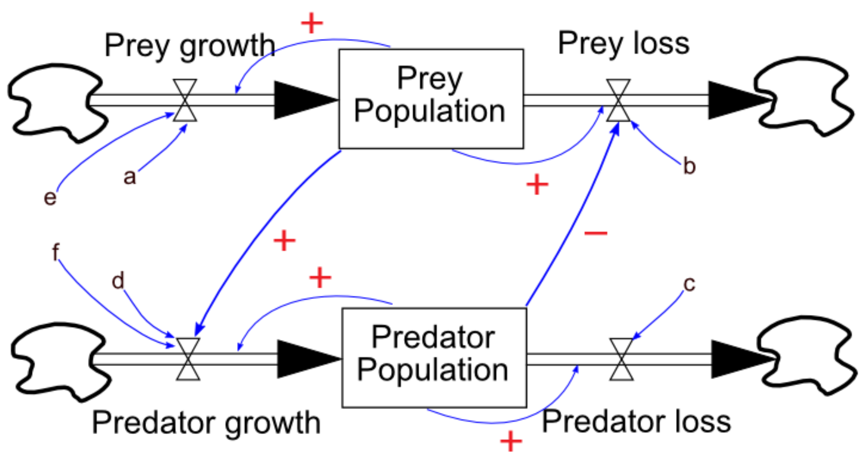


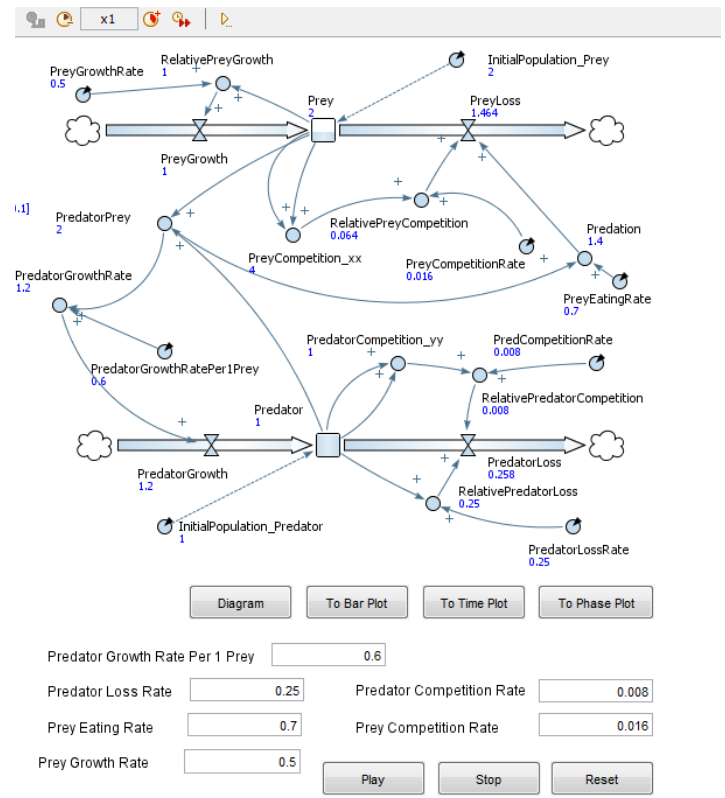
Stock and Flow Diagram

Model Modification

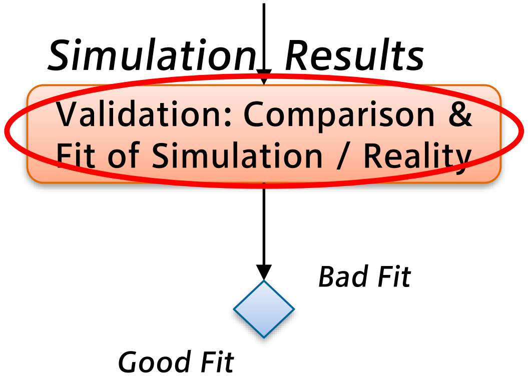


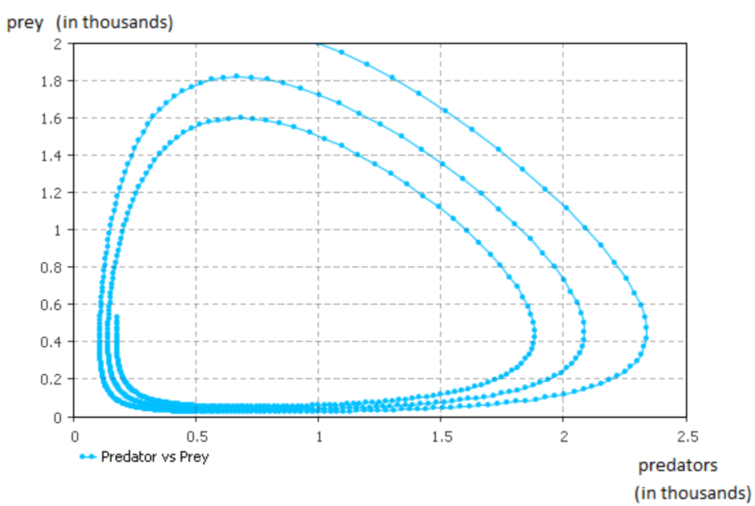
Stock and Flow Diagram





Model Modification





Parameters:

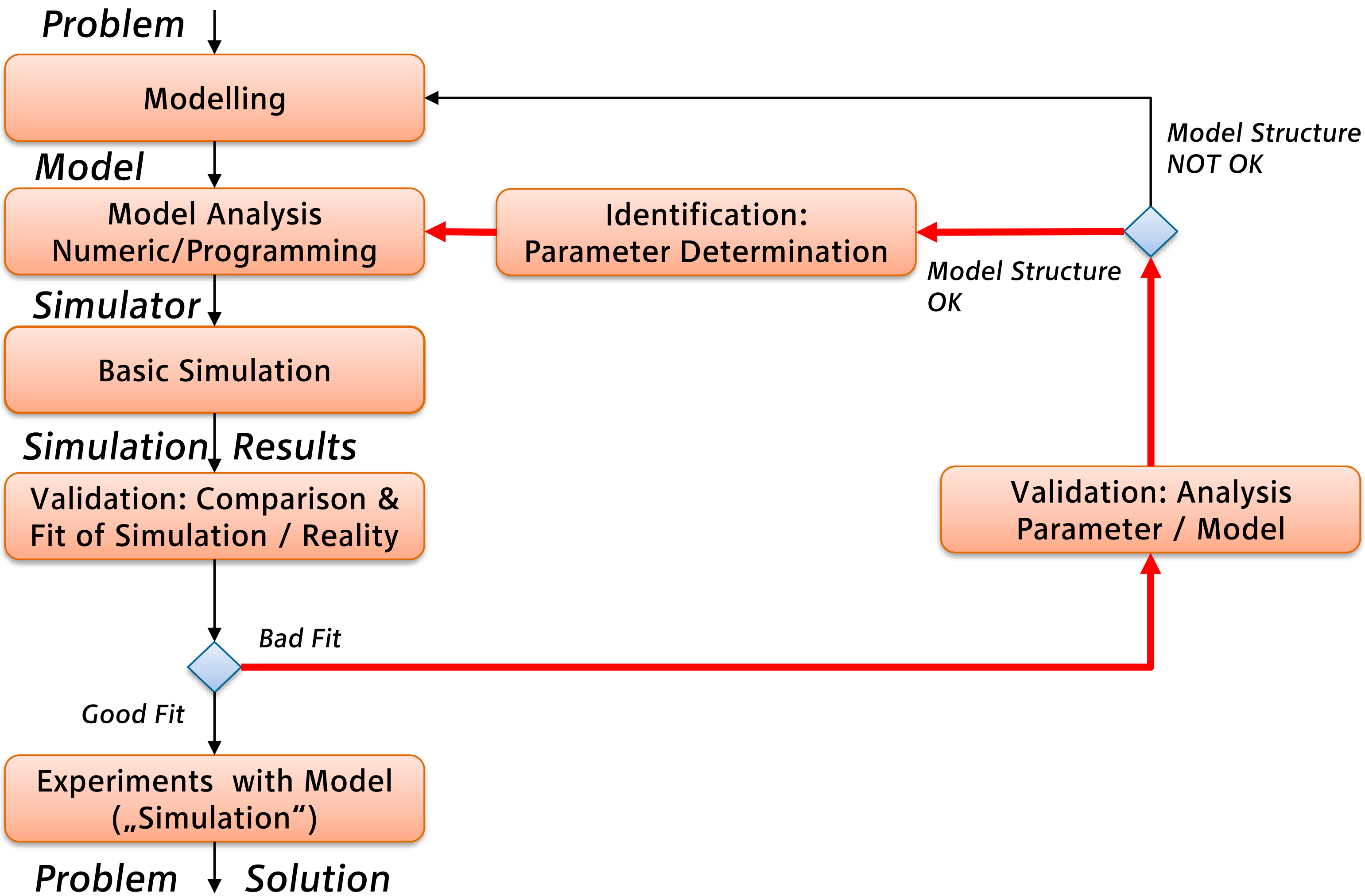
Predator Growth Rate Per 1 Prey	0.6
Predator Loss Rate	0.25
Prey Eating Rate	0.7
Prey Growth Rate	0.5
Predator Competition Rate	0.0080
Prey Competition Rate	0.016

$$\dot{x} = (a - b\cdot y)x - e\cdot x^2 = (a - e\cdot x - b\cdot y)x$$

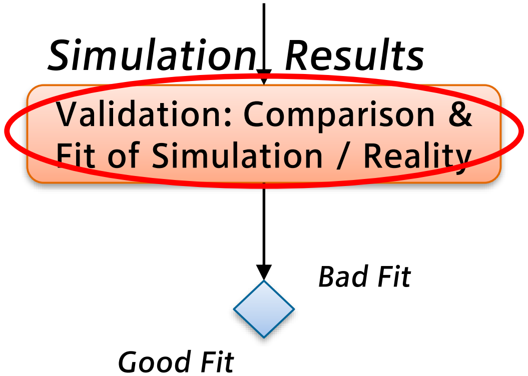
$$\dot{y} = (-c + d\cdot x)y - f\cdot y^2 = (-c - f\cdot y + d\cdot x)y$$

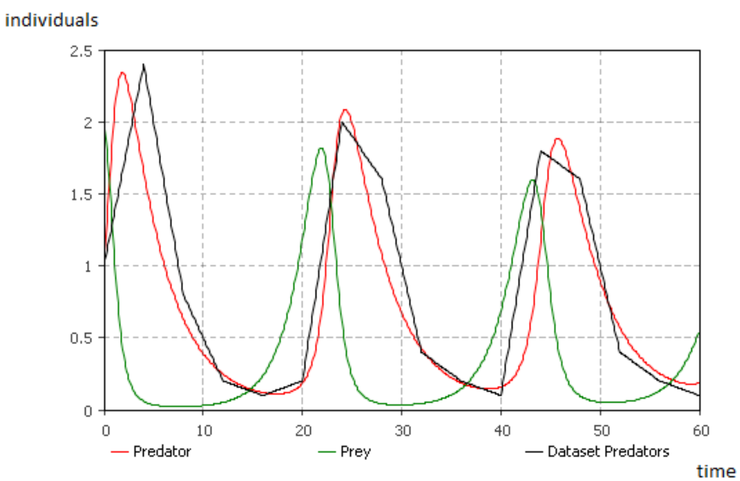
Simulation Circle: Predator - Prey

Mathematical Modelling and Simulation



Model Modification





Parameters:

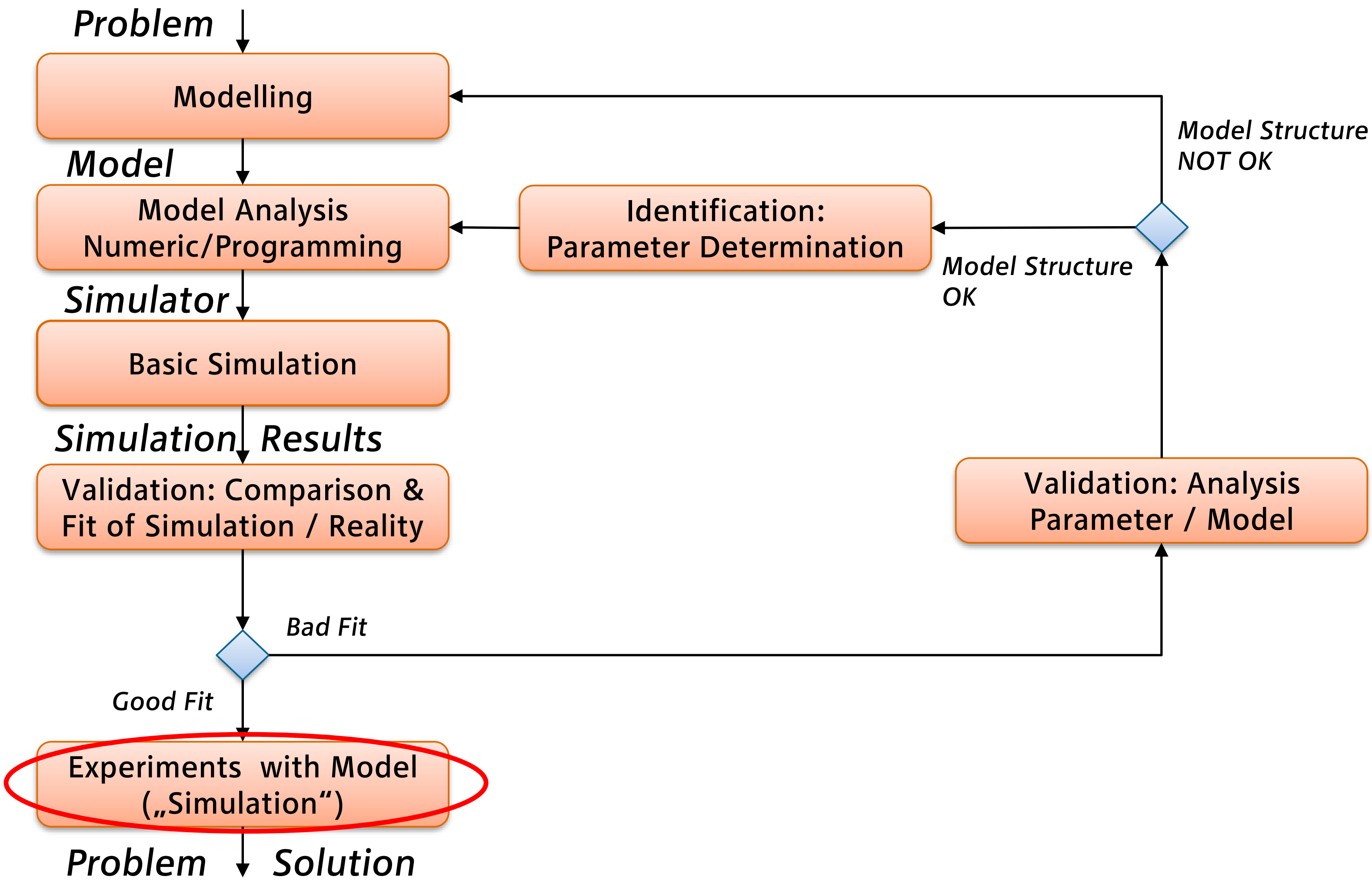
Predator Growth Rate Per 1 Prey	0.6
Predator Loss Rate	0.25
Prey Eating Rate	0.7
Prey Growth Rate	0.5
Predator Competition Rate	0.0080
Prey Competition Rate	0.016

$$\dot{x} = (a - b \cdot y)x - e \cdot x^2 = (a - e \cdot x - b \cdot y)x$$

$$\dot{y} = (-c + d \cdot x)y - f \cdot y^2 = (-c - f \cdot y + d \cdot x)y$$

Simulation Circle: Predator - Prey

Mathematical Modelling and Simulation



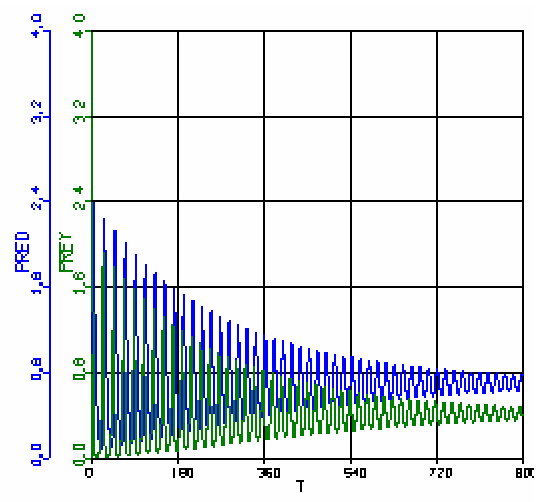
Results Interpretation / Analysis

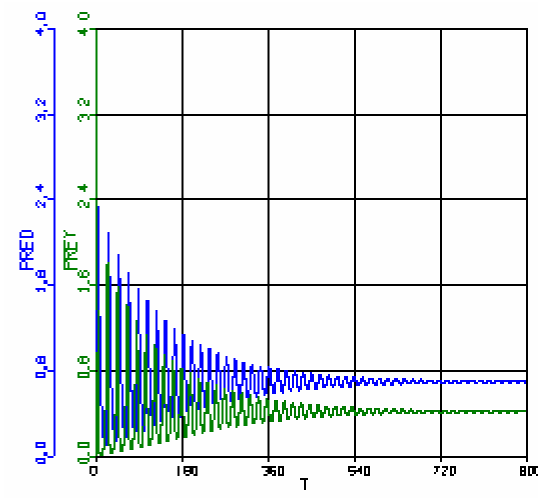
Good Fit

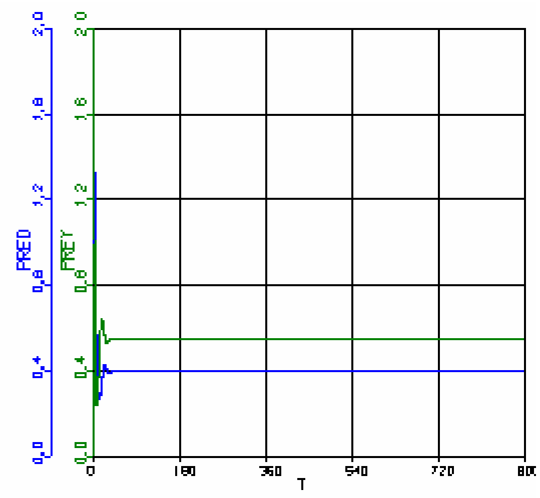
Experiments with Model („Simulation“)

Problem *Solution*

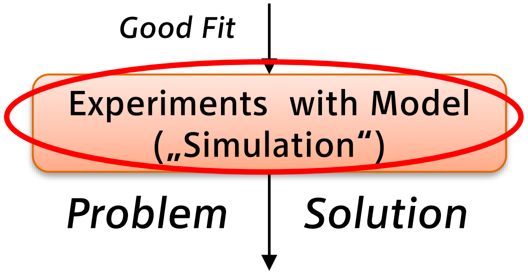
- Determination of long time behavior / stationary solutions (equilibria)



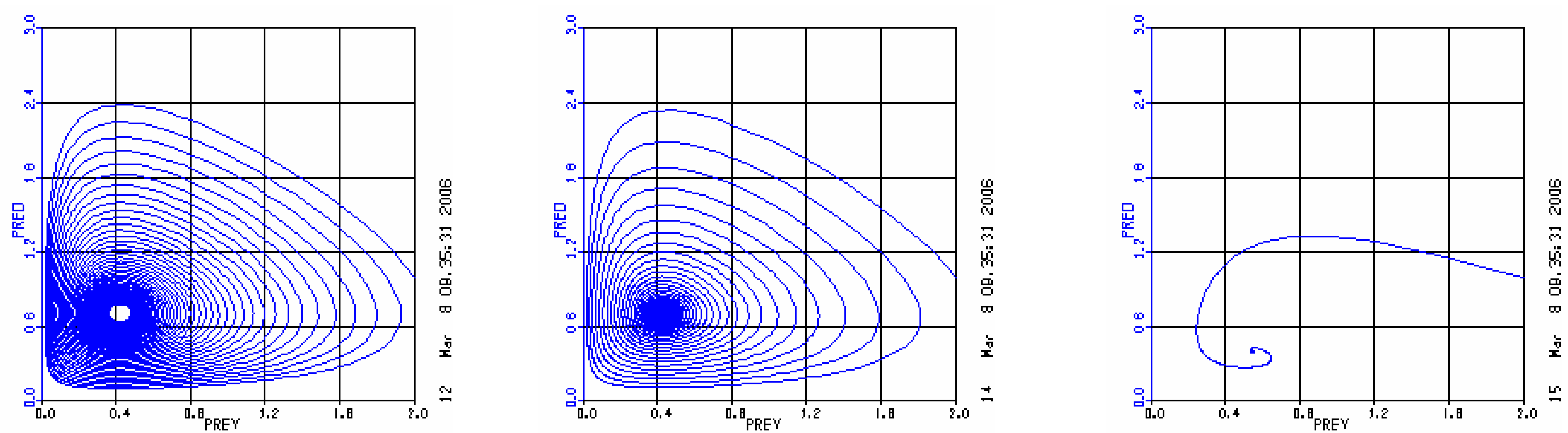




Results Interpretation / Analysis



- Determination of long time behavior / stationary solutions (equilibria)



Use of Pesticide

Good Fit

Experiments with Model („Simulation“)

Problem *Solution*

Modification of Predator-prey model with intraspecific competition

- Assume, that at a specific time poison is released into the system, e.g. some of predators are removed
  from the population by hunting.

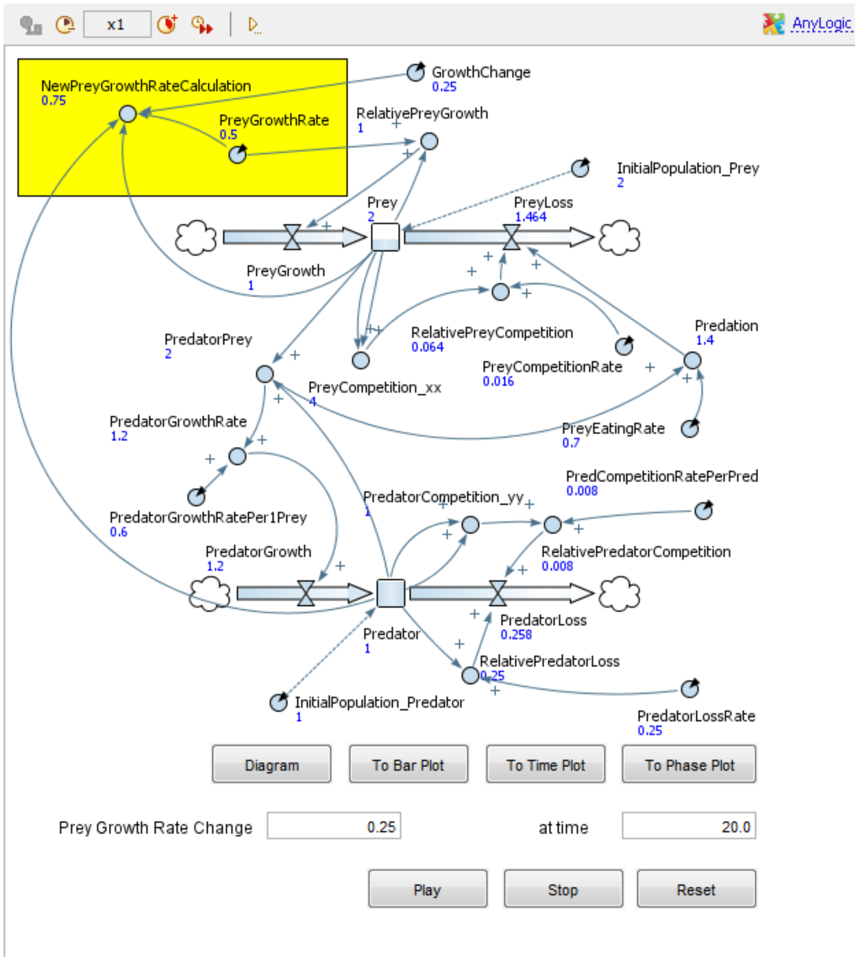
- The growth rate **a** of prey is changed to:
  where K is growth rate change.

- This change occurs at the specific time point.

- The new growth rate **a** depends on the difference between populations at this specific time point and stays constant after that.

Use of Pesticide

Mathematical Modelling and Simulation



Adequate time instant

Good Fit

Experiments with Model („Simulation“)

Problem *Solution*

$$t_c : \quad d_{old} \rightarrow d_{new}, \quad f_{old} \rightarrow f_{new}$$

$$\dot{x} = ax - bxy - ex^2$$
$$\dot{y} = -cy + dxy - fy^2$$

$$d_{neu} = d_{alt} + d_c(x(t_c) - y(t_c))$$
$$f_{neu} = f_c \, f_{alt}$$

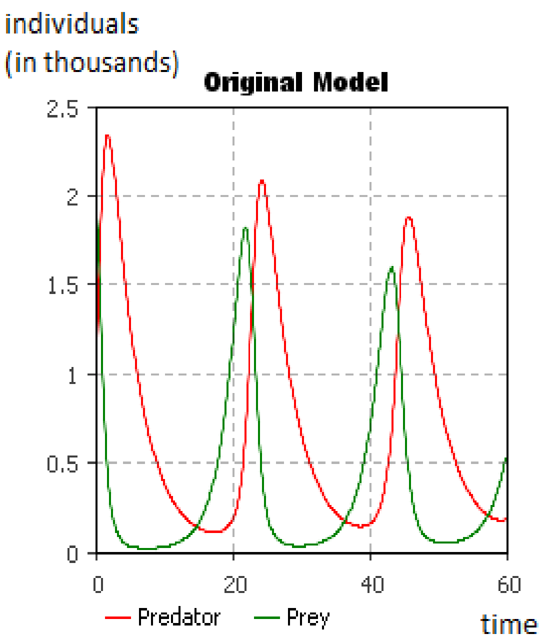
Modification of Predator-prey model with intraspecific competition

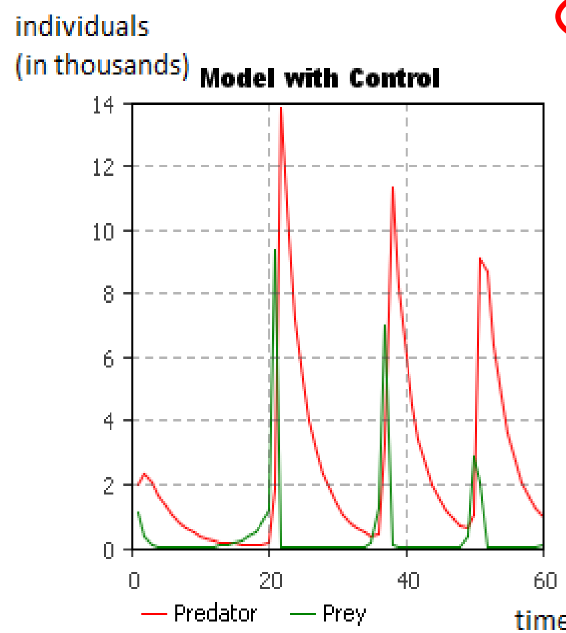
Good Fit

Experiments with Model („Simulation“)

Problem *Solution*

Population development over time:





Parameters:

Pred Loss Rate Change 2.0

at time 20.0

Note: Please note the different scaling of the plots.

$$d_{neu} = d_{alt} + d_c(x(t_c) - y(t_c))$$

$$f_{neu} = f_c \, f_{alt}$$

Modification of Predator-prey model with intraspecific competition

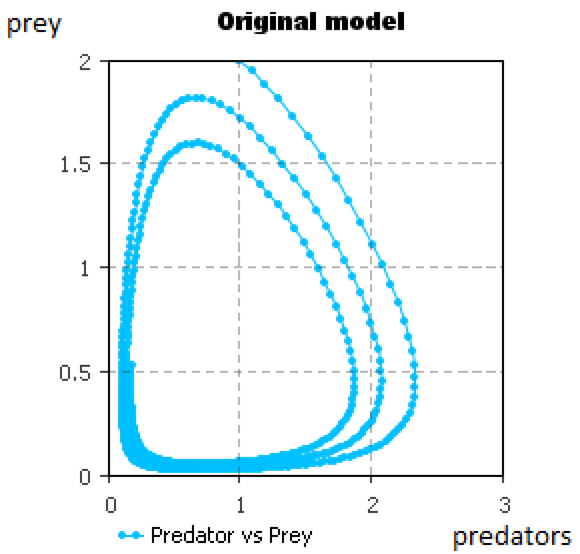
Mathematical Modelling and Simulation

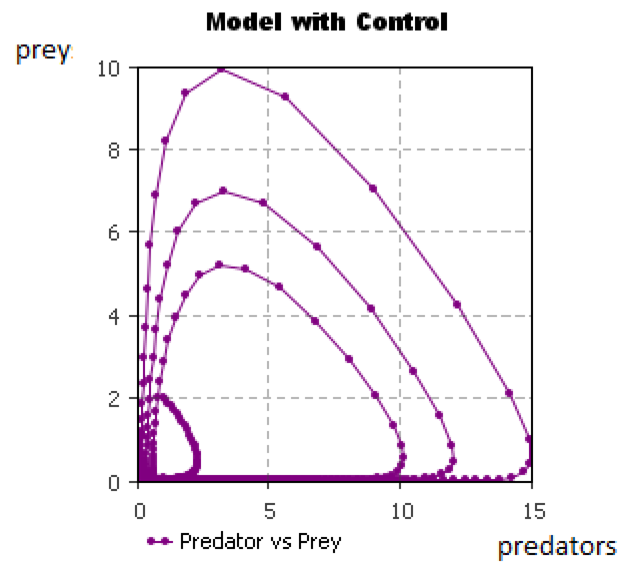
Good Fit

Experiments with Model („Simulation“)

Problem ***Solution***

Population development over time:





Parameters:

Pred Loss Rate Change 2.0

at time 20.0

$$d_{neu} = d_{alt} + d_c(x(t_c) - y(t_c))$$

$$f_{neu} = f_c\, f_{alt}$$

Modification of Predator-prey model with intrasp‌ecific competition

Mathematical Modelling and Simulation

Problem

Modelling

Model

Good Fit

Experiments with Model („Simulation“)

Problem | *Solution*

Dynamics: Prey – Predators

Environment: isolated

Measurement: natural enemies

5 Years = 60 months

quarterly

Assignment: short time, changes the growth of preys, damping parameter

Approach: optimal time point t_c is dependent on the population difference

Problem: When is a reasonable time to use chemical pesticides?

Result: **The assignment is not conducive**

The DON'Ts of Mathematical Modelling

(S. W. Golomb, Simulation 14 (1970), 197-198)

- DON'T believe that the model is the reality
- DON'T extrapolate beyond the region of fit
- DON'T distort reality to fit the model
- DON'T retain a discredited model
- DON'T fall in love with your model

Introduction to System Dynamics

Overview

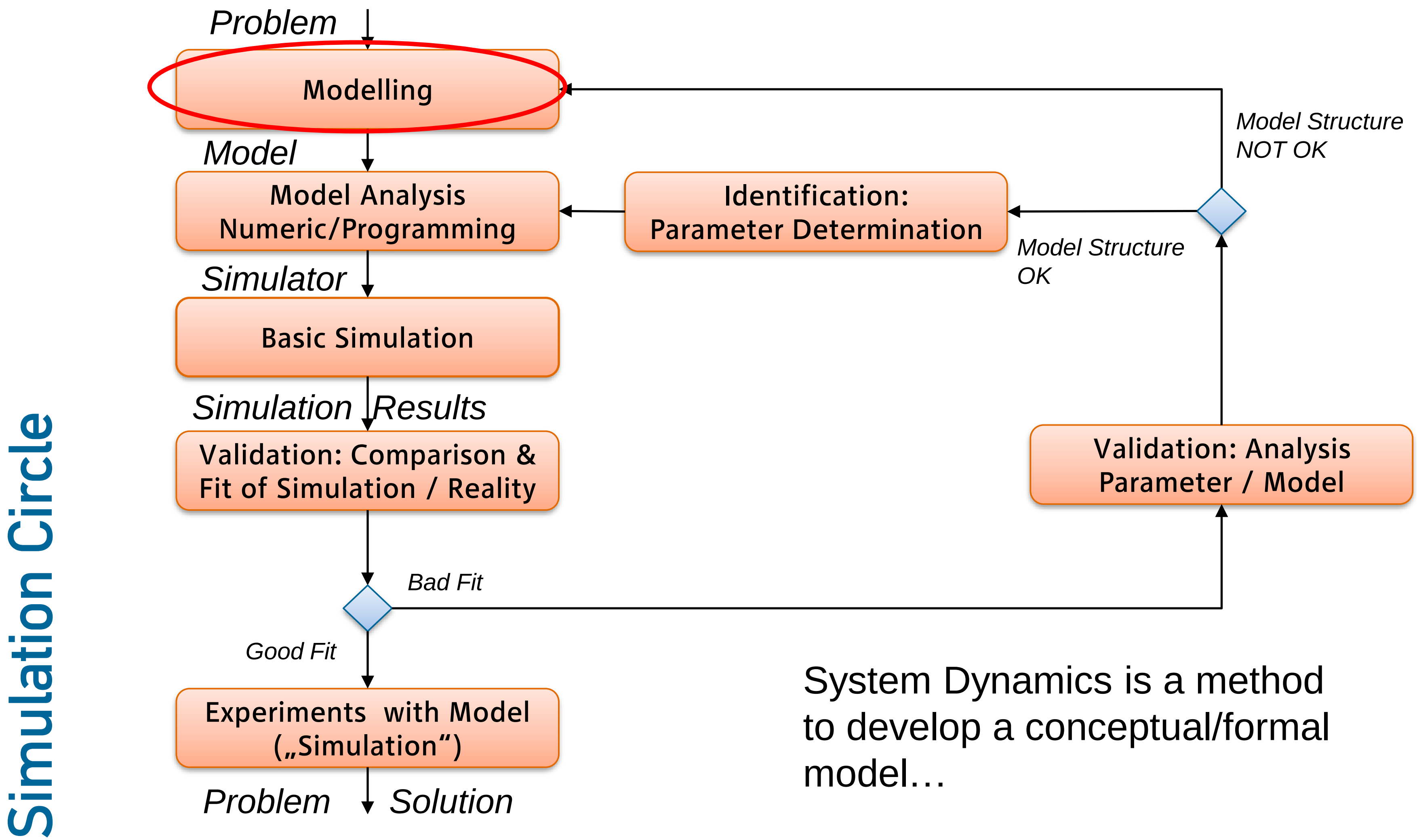
- Introduction
- General Information
- How to Build a System Dynamics Model
  - System Variables and Boundaries
  - Causal Loop Diagrams
  - Stock-and-Flow Diagrams
- Helpful Tools
- Analysis
- Simulators
- Conclusion
- Further Steps

General Information (1)

Mathematical Modelling and Simulation

- System Dynamics (short SD) is a modelling and simulation method developed by **Jay W. Forrester**.
- He adapted methods formerly used for system analysis of technological systems to social systems (MIT Sloan School of Management, 1956).
- Thus he was criticising mathematical models developed for management sciences.
- SD has roots on **control theory** and **nonlinear dynamics**
- SD is very intuitive, supported by graphics

General Information (2)

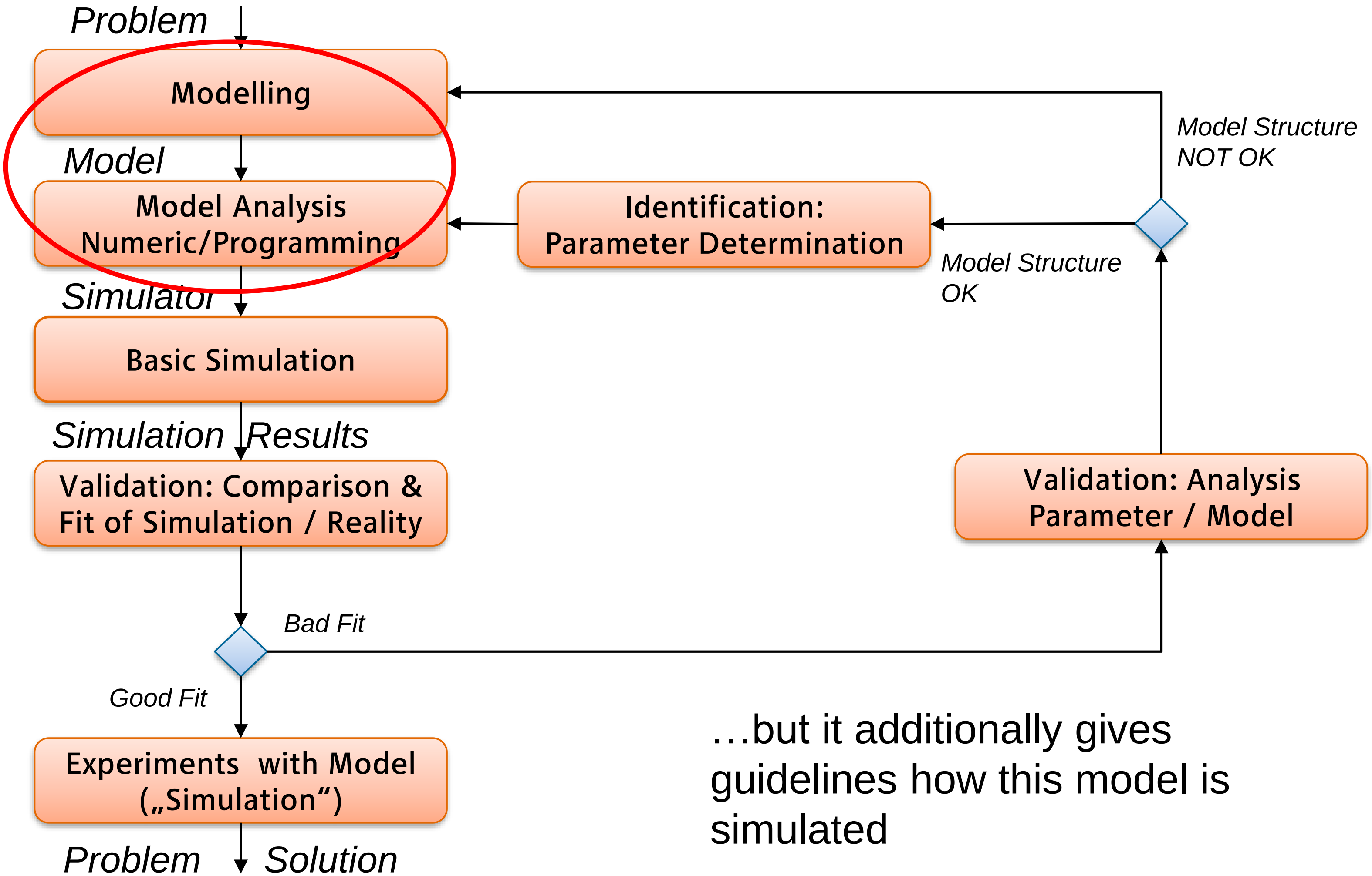


System Dynamics is a method to develop a conceptual/formal model…

General Information (2)

Mathematical Modelling and Simulation

Simulation Circle



…but it additionally gives guidelines how this model is simulated

General Information (3)

<u>Hypothesis</u>:

- Manager usually know very good about processes and their causal relationships within their companies (system).
- The behaviour of a system is mostly predetermined by its (complex) structure.
- Practically useful models can usually not be simulated by analytic calculations.

General Information (4)

Mathematical Modelling and Simulation

Literature:

- 1961: *Industrial Dynamics* (Forrester)
- 1969: *Urban Dynamics* (Forrester), first use of System Dynamics apart from economic businesses.
- 1970: *World Dynamics* (Forrester), superwised by Club of Rome, use of System Dynamics for development of a so called „World Model“. Similar:
- 1972: Meadows et al.: *The Limits to Growth*

General Information (5)

<u>Relationship: SD & Differential Equations Modelling</u>

- Each System Dynamics model is equvalent to exactly one differential-equation (DE) system. It can be seen to be a graphical way for development of DE models.
- Advantages:
  - Picturesque
  - Optimized to understand dynamics and causal relationships of the system.
  - Finally calculated like a DE model.

General Information (5)

Relationship: SD & Differential Equations Modelling

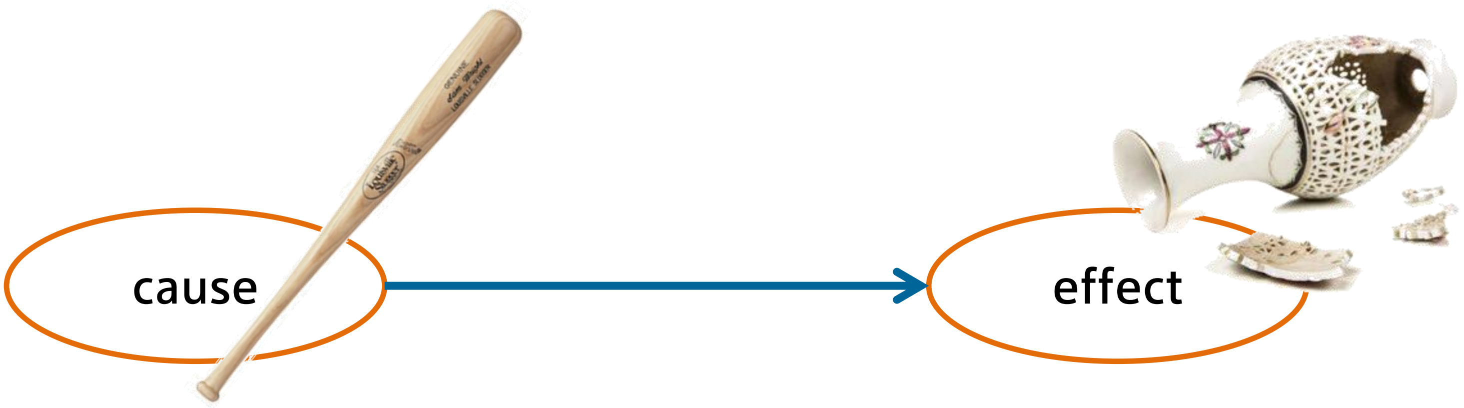
- Each System Dynamics model is equvalent to exactly one differential-equation (DE) system. It can be seen to be a graphical way for development of DE models.
- Advantages:
  - Picturesque
  - Optimized to understand dynamics and causal relationships of the system.
  - Finally calculated like a DE model

Perfect starting-point for learning about Modelling and Simulation

Key to develop SD Models

Causal thinking is the key to organizing ideas in a system dynamics study

(Roberts et al. 1983)



How to build a SD Model?

1. Identify system variables and system boundaries

6

2. Capture **links** of variables in a **Causal Loop Diagram (CLD)**
3. Build a **Stock and Flow Diagram (SFD)**

- Implement the model in a simulator

1. System Variables and Boundaries

Mathematical Modelling and Simulation

a. Analysis of the problem - Determining the purpose and the use of the model and defining a target for the simulation.

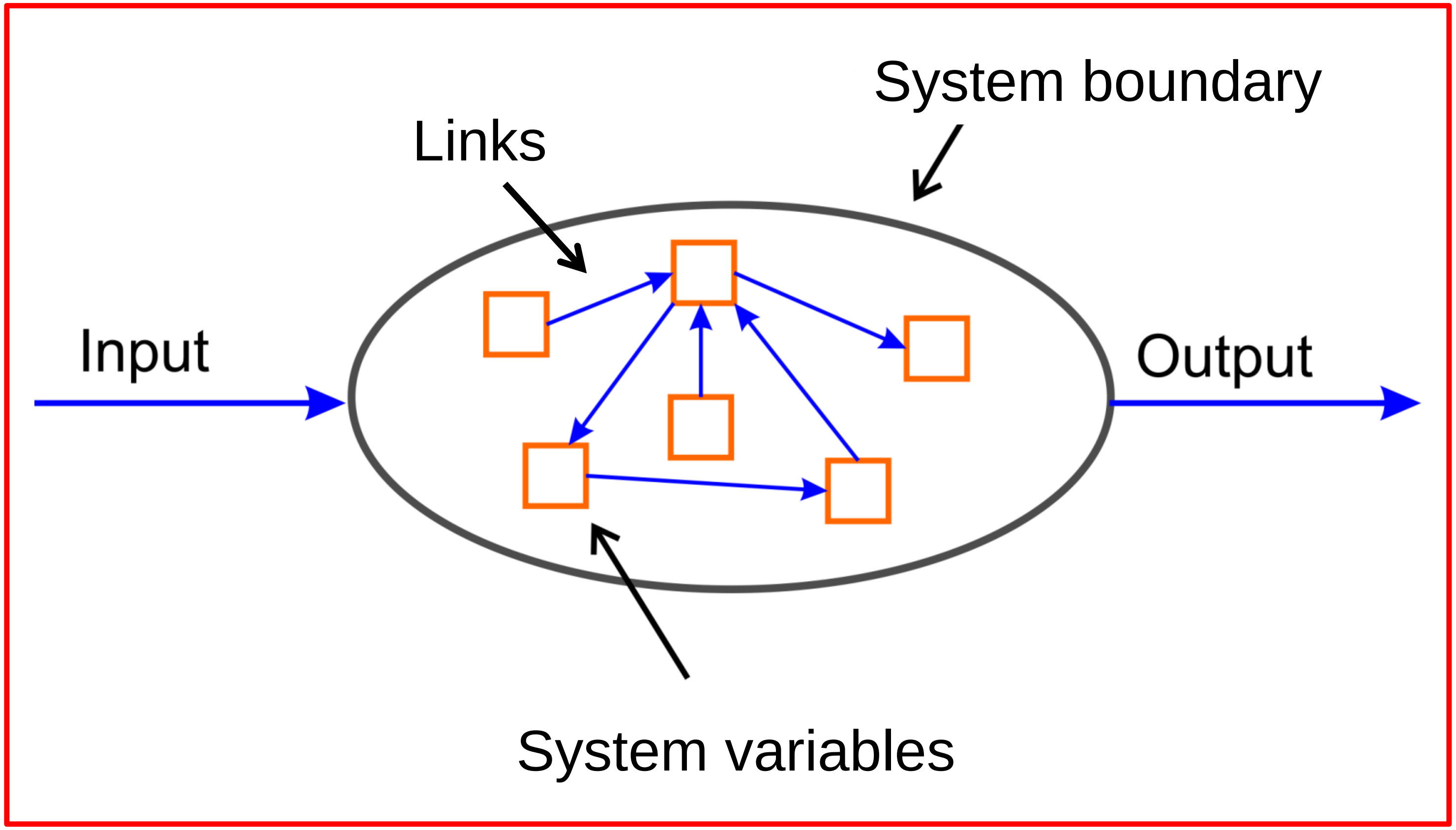
b. Start collecting information and data. Start developing hypothesis about the parts of the system.

c. Determine the elements of the system.

d. Determine causal relationships between the elements.

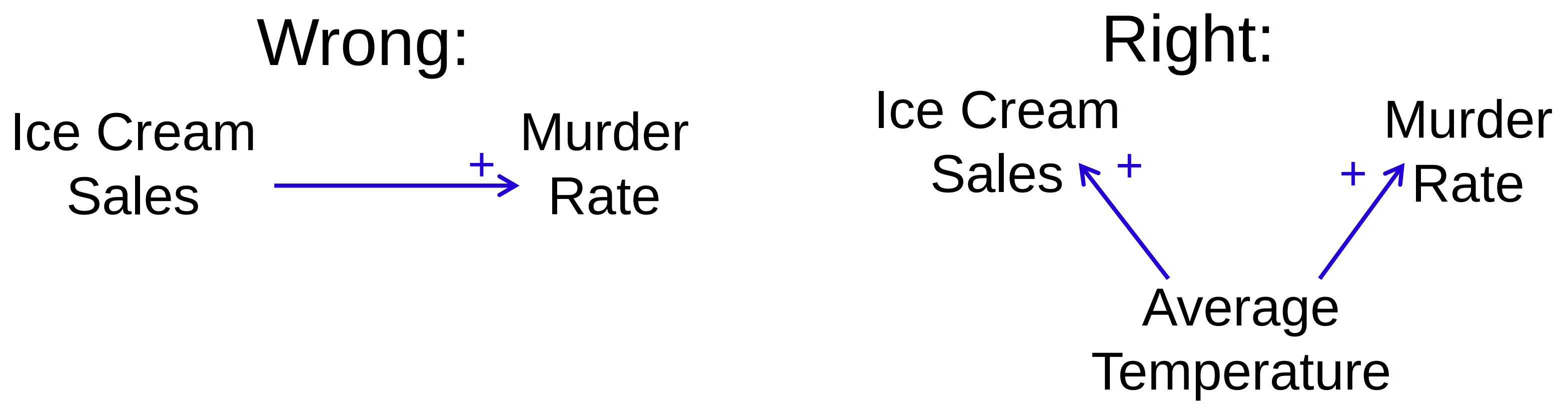
1. System Variables and Boundaries

Mathematical Modelling and Simulation



Causation vs. Correlation

- ***Correlation*** represents past behavior and not the structure of the system
- ***Causation*** represents the causal links of the structure



Causality vs Correlation

Mathematical Modelling and Simulation

<u>Causation vs. Correlation</u>

- ***Correlation*** represents past behavior and not the structure of the system
- ***Causation*** represents the causal links of the structure

Correlation	Wrong Causal Implication	Lesson?
Smoking, Lung Cancer (+)	People suffering from lung cancer are more likely to start smoking	??

Causality vs Correlation

Mathematical Modelling and Simulation

Causation vs. Correlation

- ***Correlation*** represents past behavior and not the structure of the system
- ***Causation*** represents the causal links of the structure

Correlation	Wrong Causal Implication	Lesson?
Smoking, Lung Cancer (+)	People suffering from lung cancer are more likely to start smoking	Causality is always directed! Be careful to take the correct one.
Darkness, Electricity Consumption (-)	If it was darker, we could reduce our energy problems	??

Causality vs Correlation

Mathematical Modelling and Simulation

Causation vs. Correlation

- ***Correlation*** represents past behavior and not the structure of the system
- ***Causation*** represents the causal links of the structure

Correlation	Wrong Causal Implication	Lesson?
Smoking, Lung Cancer (+)	People suffering from lung cancer are more likely to start smoking	Causality is always directed! Be careful to take the correct one.
Darkness, Electricity Consumption (-)	If it was darker, we could reduce our energy problems	Always look for direct causalities! Don't foget that people sleep when its dark...
Murder Rate, Ice Cream Sales (+)	Ice cream makes people potential murderes	??

Causality vs Correlation

Mathematical Modelling and Simulation

Causation vs. Correlation

- ***Correlation*** represents past behavior and not the structure of the system
- ***Causation*** represents the causal links of the structure

Correlation	Wrong Causal Implication	Lesson?
Smoking, Lung Cancer (+)	People suffering from lung cancer are more likely to start smoking	Causality is always directed! Be careful to take the correct one.
Darkness, Electricity Consumption (-)	If it was darker, we could reduce our energy problems	Always look for direct causalities! Don't foget that people sleep when its dark...
Murder Rate, Ice Cream Sales (+)	Ice cream makes people potential murderes	Always look for confounding factors! E.g. the average Temperature?

Causality vs Correlation

Mathematical Modelling and Simulation

Causation vs. Correlation

- ***Correlation*** represents past behavior and not the structure of the system
- ***Causation*** represents the causal links of the structure

Famous example (1):

The NEW ENGLAND JOURNAL of MEDICINE

OCCASIONAL NOTES

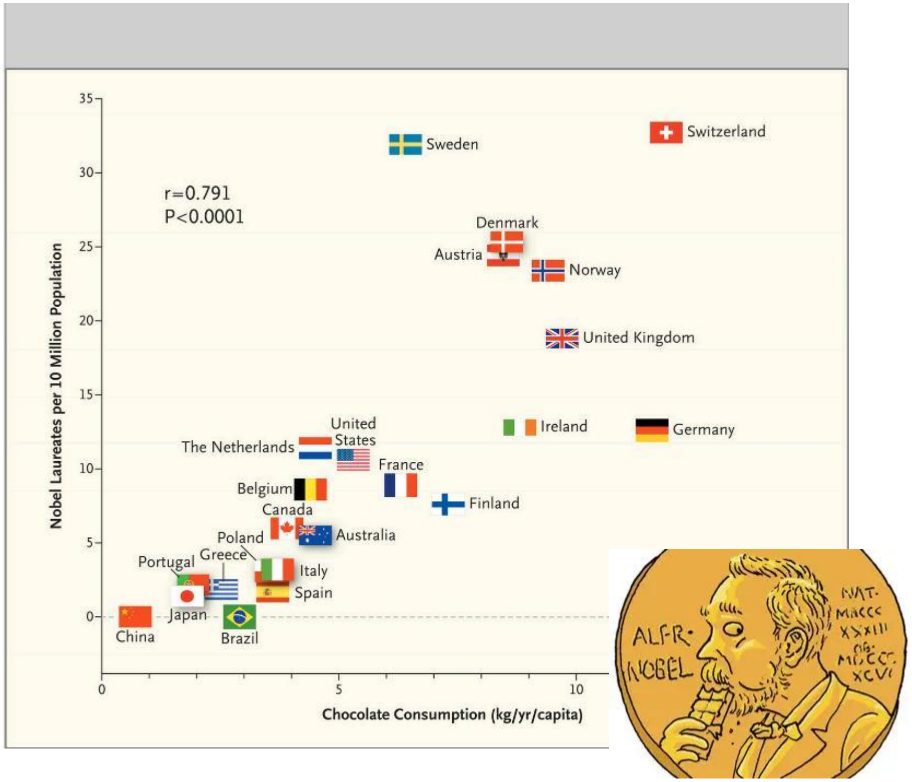
Chocolate Consumption, Cognitive Function, and Nobel Laureates

Franz H. Messerli, M.D.

Dietary flavonoids, abundant in plant-based foods, have been shown to improve cognitive function. Specifically, a reduction in the risk of dementia, enhanced performance on some cognitive tests, and improved cognitive function in elderly patients with mild impairment have been associated with a regular intake of flavonoids.[1,2] A subclass of flavonoids called flavanols which are widely

cause the population of a country is substantially higher than its number of Nobel laureates, the numbers had to be multiplied by 10 million. Thus, the numbers must be read as the number of Nobel laureates for every 10 million persons in a given country.

All Nobel Prizes that were awarded through October 10, 2011 were included. Data on per



Causality vs Correlation

Mathematical Modelling and Simulation

Causation vs. Correlation

- ***Correlation*** represents past behavior and not the structure of the system
- ***Causation*** represents the causal links of the structure

Famous example (2):

Survival in Academy Award–Winning Actors and Actresses

Donald A. Redelmeier, MD, and Sheldon M. Singh, BSc

Background: Social status is an important predictor of poor health. Most studies of this issue have focused on the lower echelons of society.

Objective: To determine whether the increase in status from winning an academy award is associated with long-term mortality among actors and actresses.

Design: Retrospective cohort analysis.

Setting: Academy of Motion Picture Arts and Sciences.

Participants: All actors and actresses ever nominated for an academy award in a leading or a supporting role were identified ($n = 762$). For each, another cast member of the same sex who was in the same film and was born in the same era was identified ($n = 887$).

Measurements: Life expectancy and all-cause mortality rates.

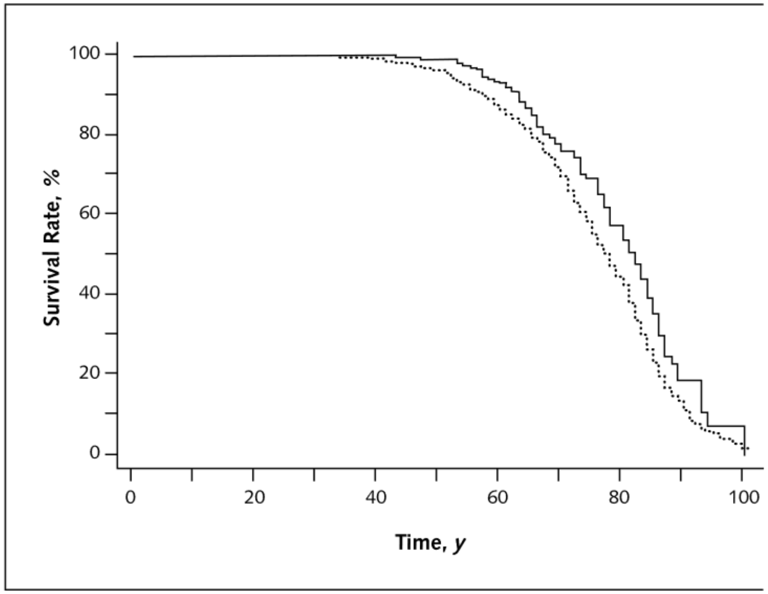
Results: All 1649 performers were analyzed; the median duration of follow-up time from birth was 66 years, and 772 deaths occurred (primarily from ischemic heart disease and malignant disease). Life expectancy was 3.9 years longer for Academy Award winners than for other, less recognized performers (79.7 vs. 75.8 years; $P = 0.003$). This difference was equal to a 28% relative reduction in death rates (95% CI, 10% to 42%). Adjustment for birth year, sex, and ethnicity yielded similar results, as did adjustments for birth country, possible name change, age at release of first film, and total films in career. Additional wins were associated with a 22% relative reduction in death rates (CI, 5% to 35%), whereas additional films and additional nominations were not associated with a significant reduction in death rates.

Conclusion: The association of high status with increased longevity that prevails in the public also extends to celebrities, contributes to a large survival advantage, and is partially explained by factors related to success.

Ann Intern Med. 2001;134:955-962. www.annals.org
For author affiliations, current addresses, and contributions, see end of text.
See editorial comment on pp 1001-1003.

Social status is a consistent, powerful, and widespread determinant of death rates. The association between high status and low mortality has appeared throughout the world, has persisted for more than a century, and

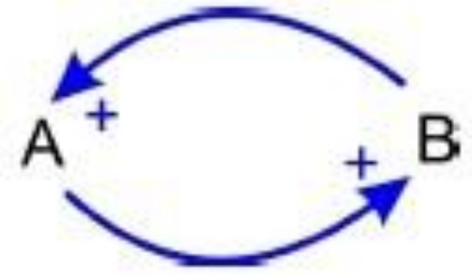
breaks to stardom are often haphazard and heavily dependent on chance. Indeed, some pundits suggest that being nominated for an Academy Award is due to talent whereas winning one is due to luck.



Analysis is based on log-rank test comparing 235 winners (99 deaths) with 887 controls (452 deaths). The total numbers of performers available for analysis were 1122 at 0 years, 1056 at 40 years, 762 at 60 years, and 240 at 80 years. $P = 0.003$ for winners vs. controls.

2. Causal Loop Diagram

Capture the **behavior** and **links** of and within the system by interlinking system variables that are related to each other



Behavior of system due to:

- Feedback Loops
- System memory (stocks)
- Delays in material and information delays

2. Causal Loop Diagram

<u>Main components of CLDs:</u>

- **System variables:** names of elements
- **Link - positive:**

Represented by a plus-sign

Increase in variable *Eating* results in an increase in variable *Weight*

2. Causal Loop Diagram

Mathematical Modelling and Simulation

Main components of CLDs:

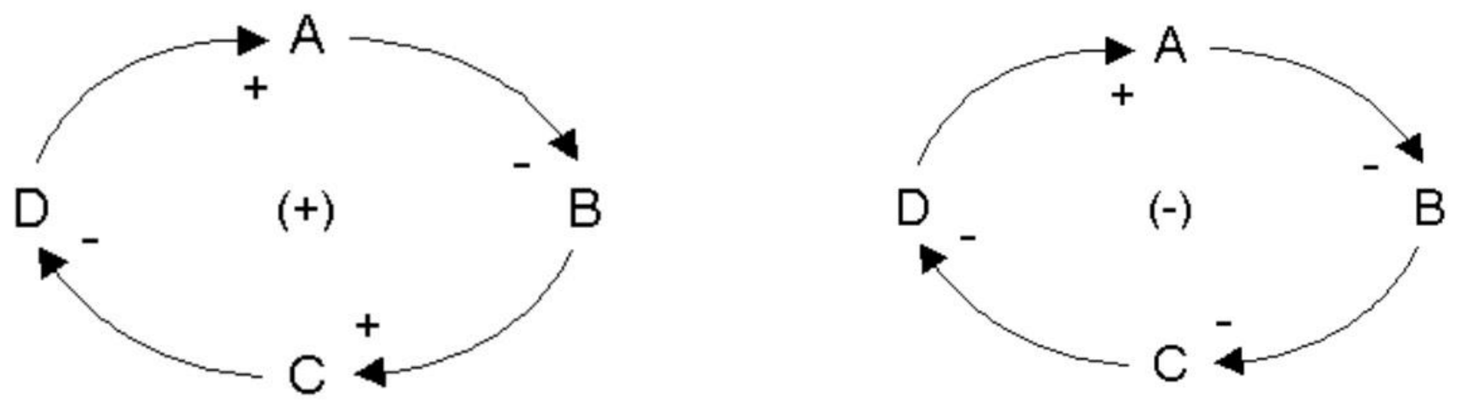
- **Link – negative:**

Represented by minus-sign.

Increase in variable *Diet* results in a decrease in variable *Weight*

Main components of CLDs:

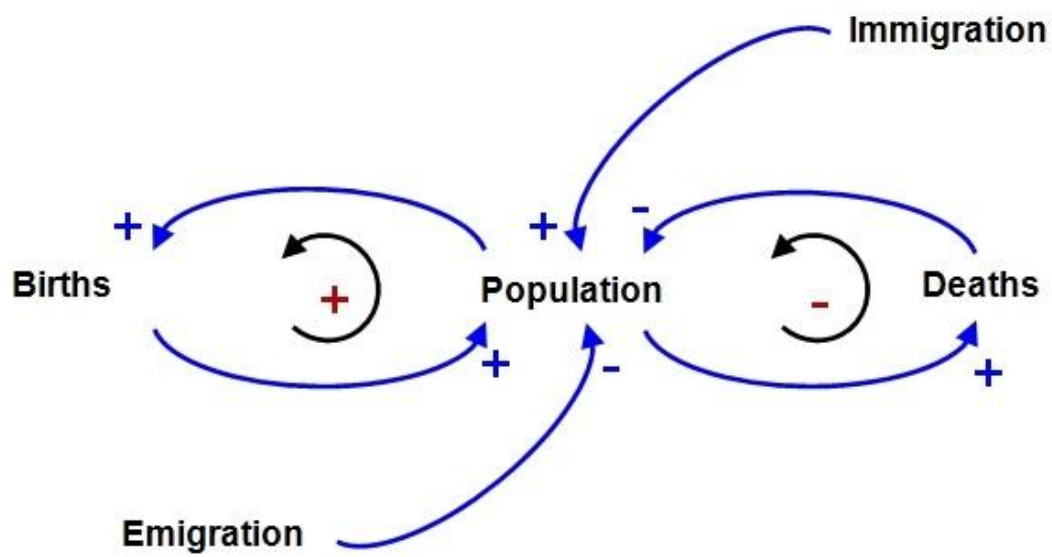
- **Feedback Loops:** are closed loops of arrows, represented by a:
“(+)” (or ”(R)” for **reinforcing**) or
“(-)” (or “(B)” for **balancing**) sign in the middle.



2. Causal Loop Diagram

<u>Main components of CLDs:</u>

- **Feedback Loops:** are closed loops of arrows, represented by a
“(+)” (or ”(R)” for **reinforcing**) or
“()” (or “(B)” for **balancing**) sign in the middle.

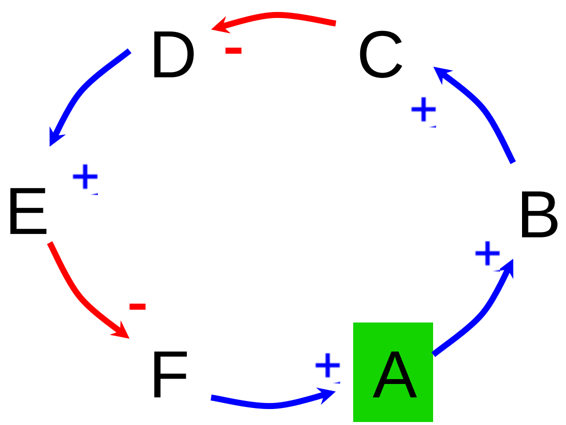


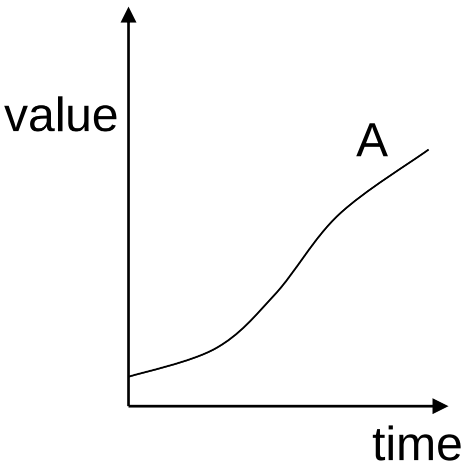
2. Causal Loop Diagram

<u>Feedback Loops:</u>

– **Reinforcing**: A system variable effects itself (via other system variable(s) of the loop), *resulting in a reinforcing of the original state* of the system variable

Even number of negative links



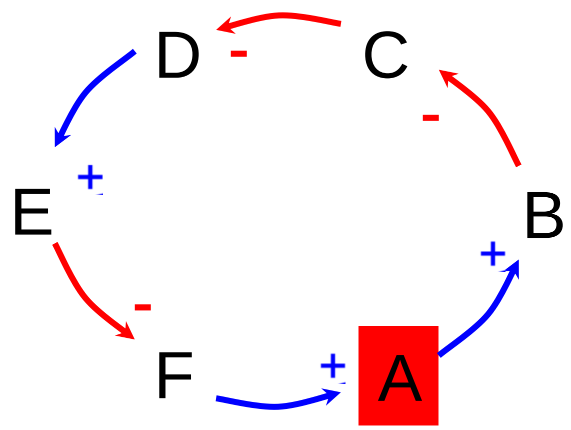


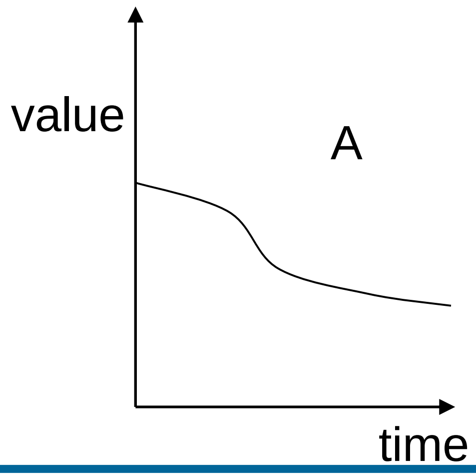
2. Causal Loop Diagram

Mathematical Modelling and Simulation

Feedback Loops:

– **Balancing:** A system variable effects itself (via other system variable(s) of the loop), resulting in a balancing of the original state of the system variable
Uneven number of negative links





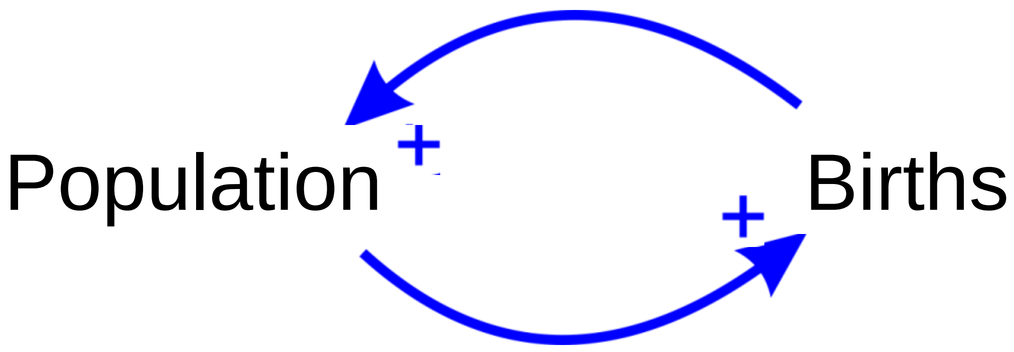
Feedback Loops

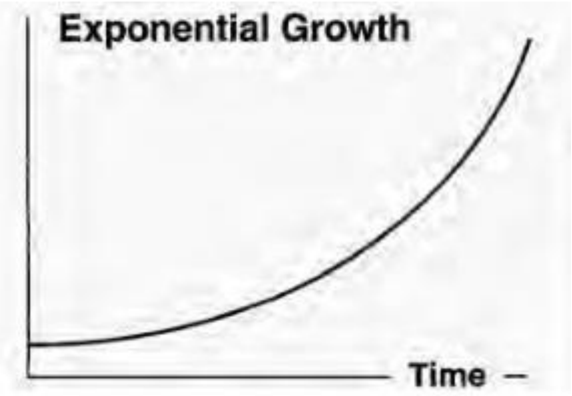
- Search to identify closed, causal feedback loops is one key element of System Dynamics
- The most important causal influences will be exactly those that are enclosed within feedback loops

Types of behavior due to loops:

- **Exponential Growth**: arises from **positive (reinforcing) feedback** loop.

Example:





2. Causal Loop Diagram

Types of behavior due to loops:

- **S-shaped Growth**: arises from a **combination of positive and negative feedback loops (nonlinear interactions)**

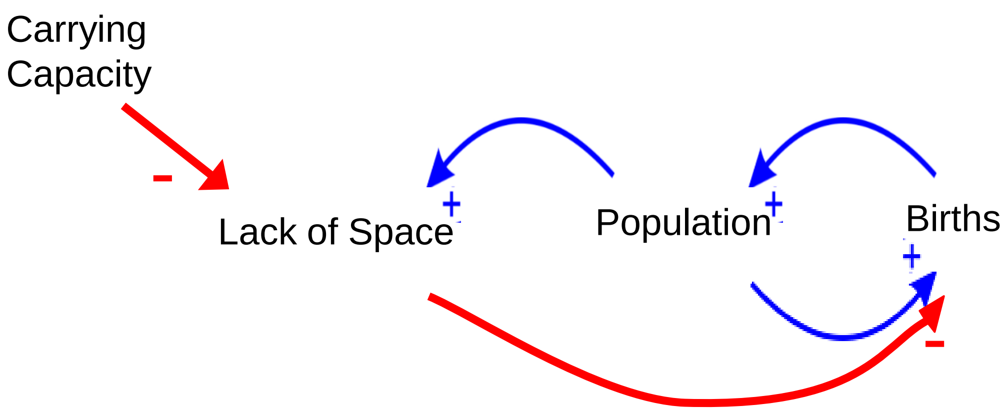
Important here:

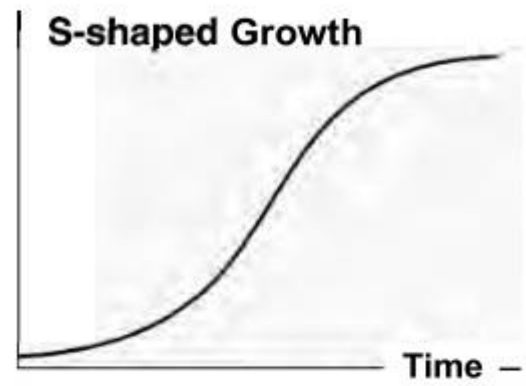
– ***Carrying capacity***: *Number of organisms a habitat can support and it is determined by the resources available in the environment and the resource requirements of the population. When the population reaches its carrying capacity the net increase rate slows down until it is zero and the population reaches its equilibrium (limit of growth)*

2. Causal Loop Diagram

<u>Types of behavior due to loops:</u>

- **S-shaped Growth**: arises from a **combination of positive and negative feedback loops (nonlinear interactions)**





2. Causal Loop Diagram

Mathematical Modelling and Simulation

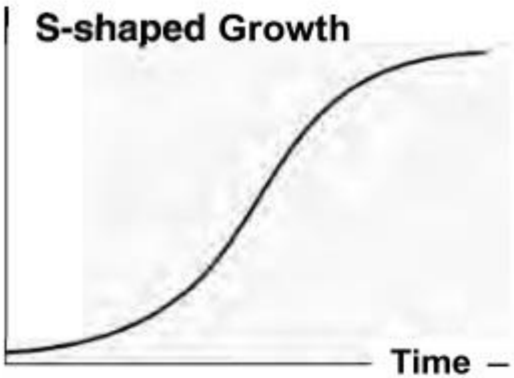
Types of behavior due to loops:

- **S-shaped Growth**: arises from a **combination of positive and negative feedback loops (nonlinear interactions)**

Necessary requirements:

– Negative feedback loops must not include any significant delays

– Carrying capacity must be fixed



2. Causal Loop Diagram

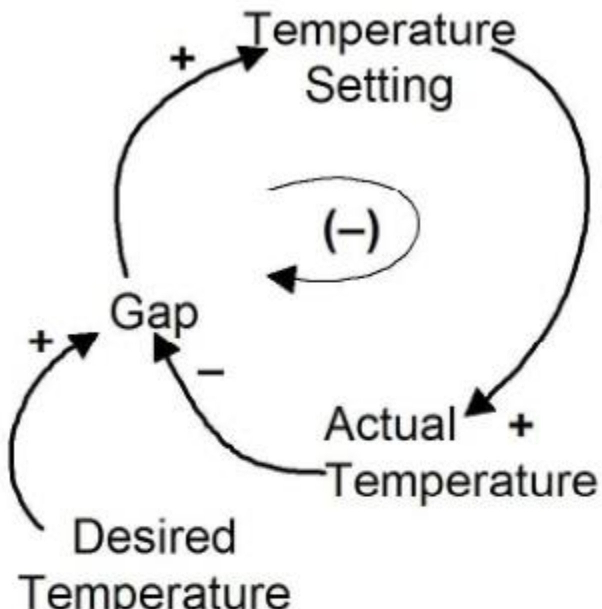
Mathematical Modelling and Simulation

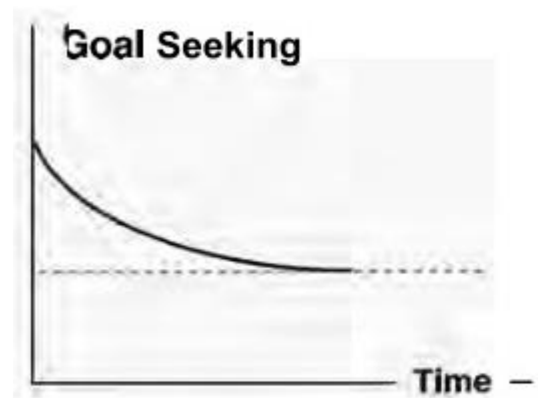
Types of behavior due to loops:

- **Goal Seeking Behavior:** arises from **negative (balancing) feedback** loop.

Example:

Corrective action



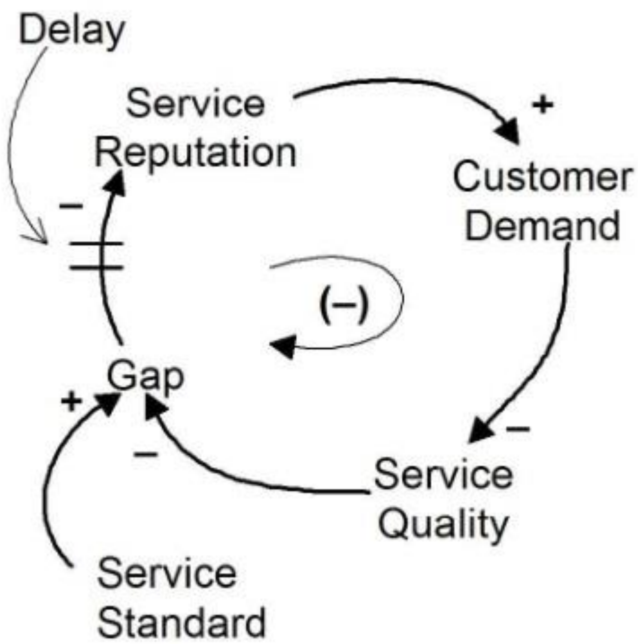


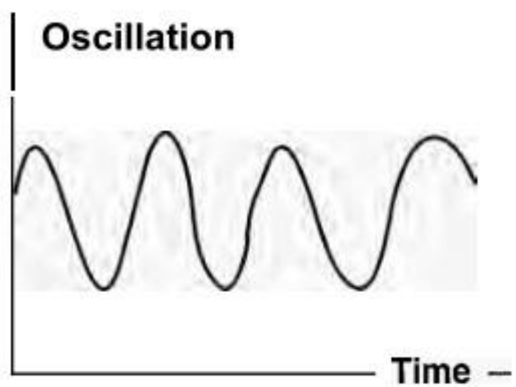
2. Causal Loop Diagram

Types of behavior due to loops:

- **Oscillation**: arises from **negative feedback with delays.**

Example:





2. Causal Loop Diagram

Types of behavior due to loops:

- **Oscillation**: arises from **negative feedback with delays.**

The state of the system is compared to the desired state of the system and corrective actions are taken. The goal is constantly overshot, then corrects / reverses and then undershoots the system and so on.

Types of behavior due to loops:

- **Oscillation**: arises from **negative feedback with delays.**

Special oscillations are:

- **Damped oscillation**: e.g. pendulum
- Chaotic oscillations

Types of behavior due to loops:

- **Oscillation**: arises from **negative feedback with delays.**

Special oscillations are:

– **Expanding oscillation and limit cycles**: If an oscillatory system is given a nudge off its equilibrium, its swings grow larger and larger until they are constrained by various nonlinearities this oscillation is called limit cycles. Predator prey populations are cycles.

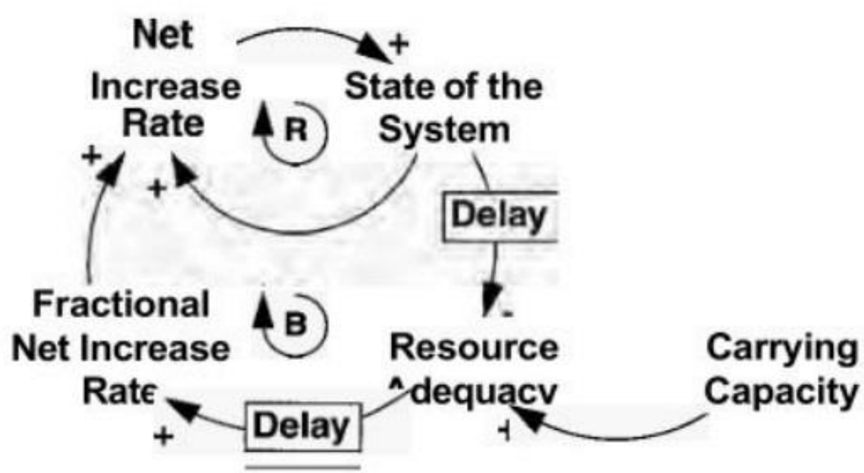
2. Causal Loop Diagram

Mathematical Modelling and Simulation

Types of behavior due to loops:

- **Growth with overshoot and oscillation**: is basically s-shaped growth with additional delay in the negative feedback loop.

Example:



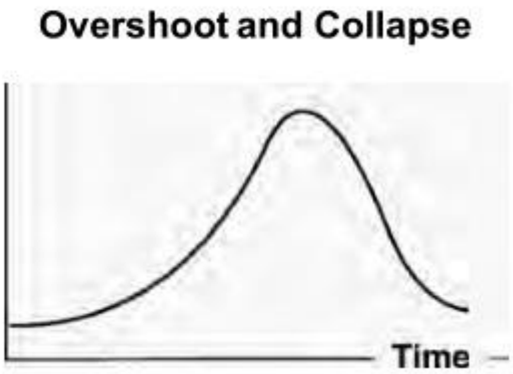
2. Causal Loop Diagram

Mathematical Modelling and Simulation

Types of behavior due to loops:

- **Overshoot and collapse**: is basically s-shaped growth but with a not fixed carrying capacity

Example: A population in a forest that grows so large, that they overbrowse the vegetation, leading to starvation and a decline in the population. If there is no regeneration of the carrying capacity, the equilibrium of the system is extinction.



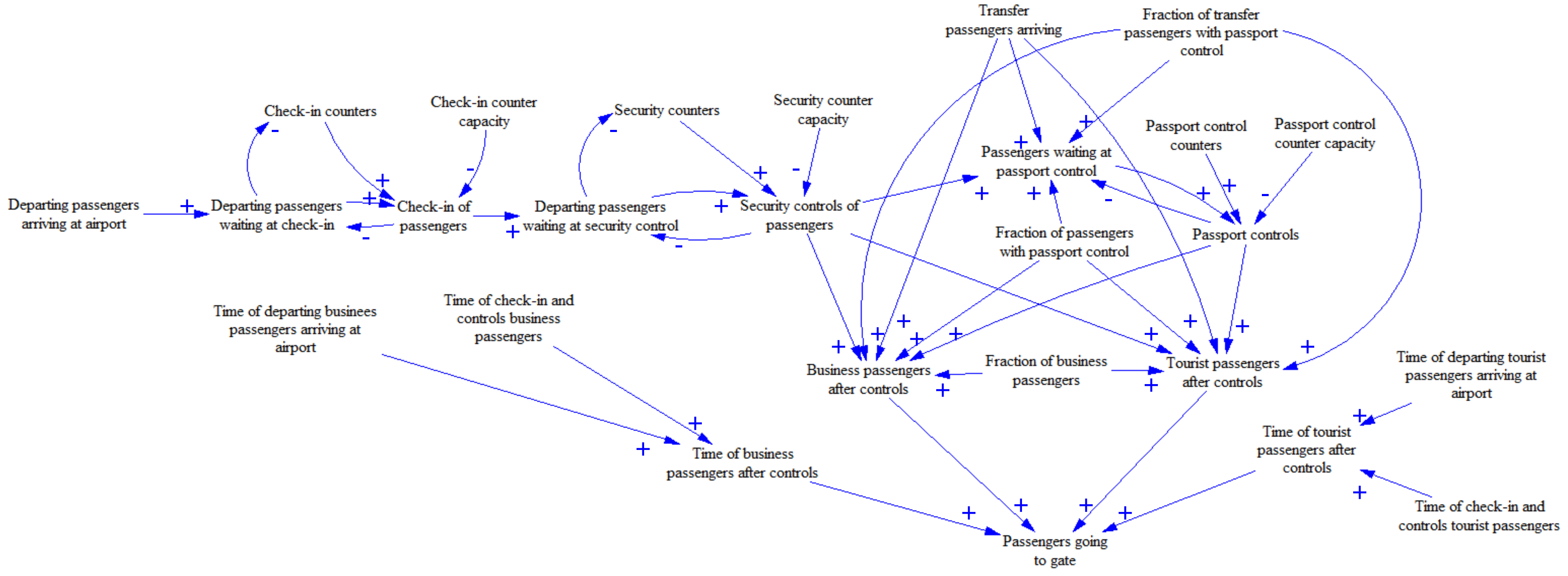
Dominating Loop

- There are systems which have more than one feedback loop within them
- A particular loop in a system of more than one loop is most responsible for the overall behavior of that system
- The dominating loop might shift over time
- When a feedback loop is within another, one loop must dominate
- Stable conditions will exist when negative loops dominate positive loops

2. Causal Loop Diagram

Mathematical Modelling and Simulation

Example:

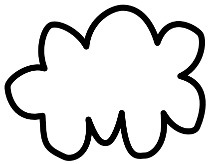


3. Stock and Flow Diagram

Problem: Not all system elements are system variables!

Solution: distinguish between

- Sources/Sinks
- Levels/Stocks
- Flows
- Auxiliaries
- Paramters
- Links



Sources/Sinks:

Source represents systems of levels and rates outside the boundary of the model

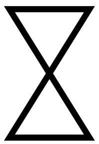
Sink is where flows terminate outside the system

E.g.: Raw Material (Source for „Construction“ Flow), Graveyard (Sink for „Dying“ Flow)

Levels/Stocks/System variables:
A quantity that accumulates over time and changes its value continuously.

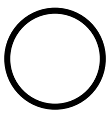
E.g.: Size of a population, Number of people waiting in a queue, Number of goods waiting to be transported, etc.

Flow/Rate/Activity/Movement:



Changes the values of levels. Every level has at least to be connected to one flow in order to change its value.

E.g.: Birth (Changes the value of the stock „population“), Eating (Changes the value of the stock „amount of food“), etc.



Auxiliary:

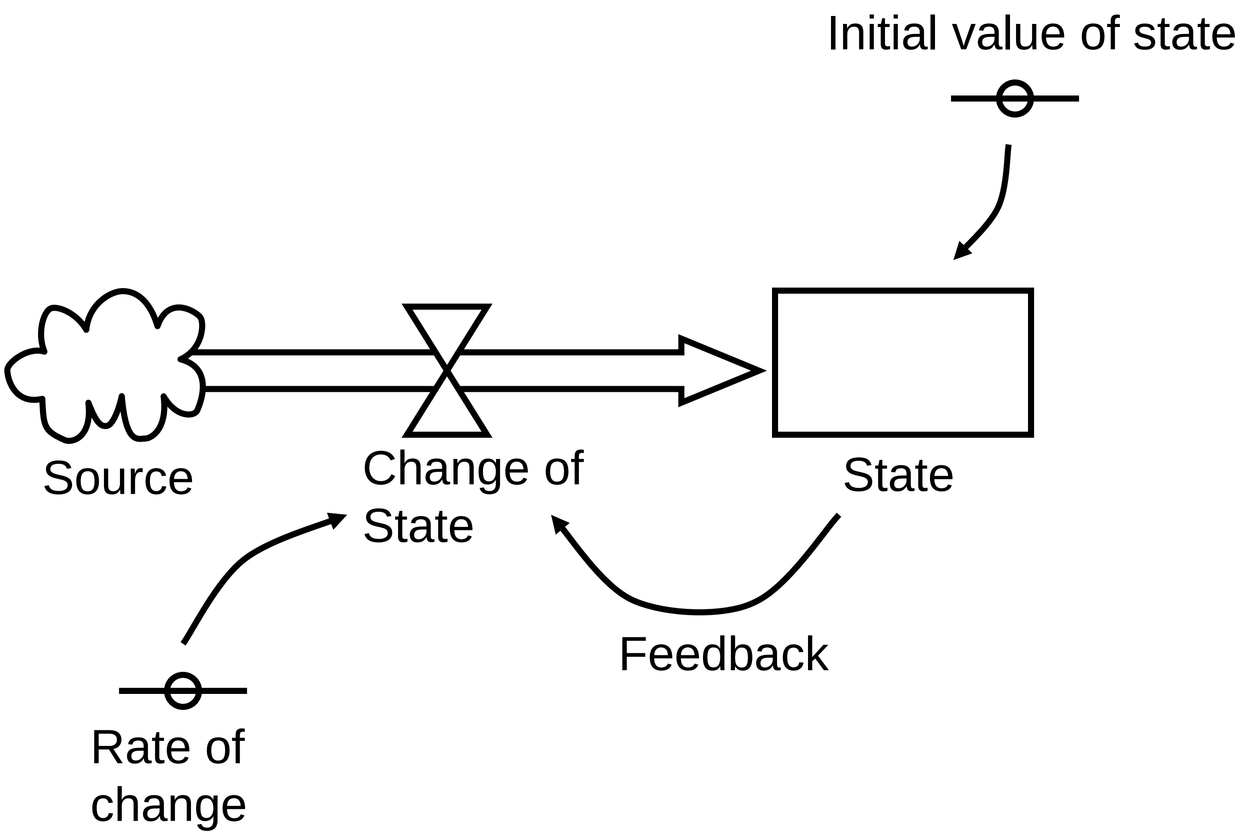
Everything that can directly/analytically be calculated out of stocks and constants. Often useful, to avoid confusing models.

E.g.: Density (can directly be calculated by the stocks/constants „mass“ and „volume“), Quelength (calculated by stock „people in queue“ and constant „average size of one person“), etc.

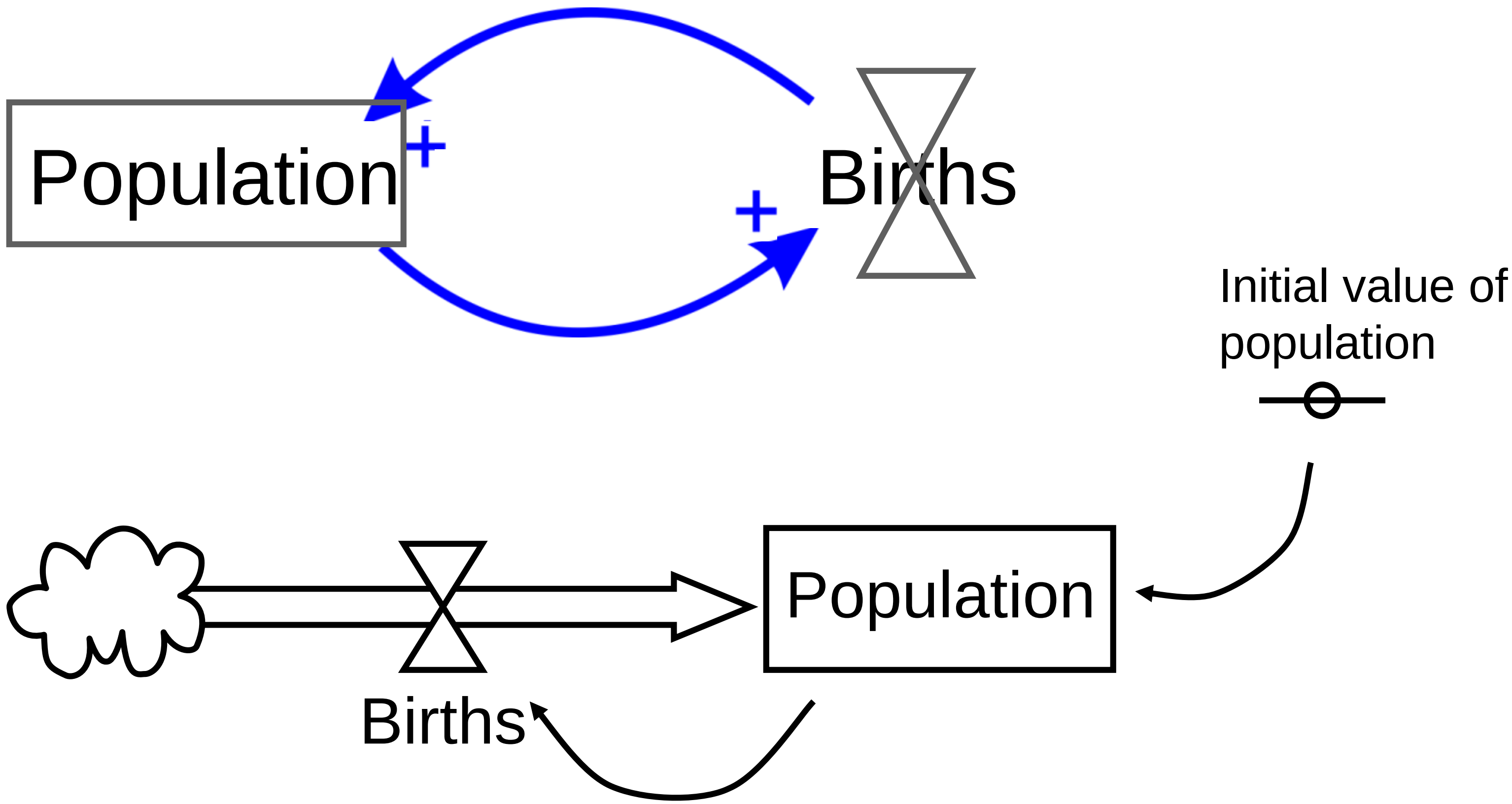
Parameter /Constant

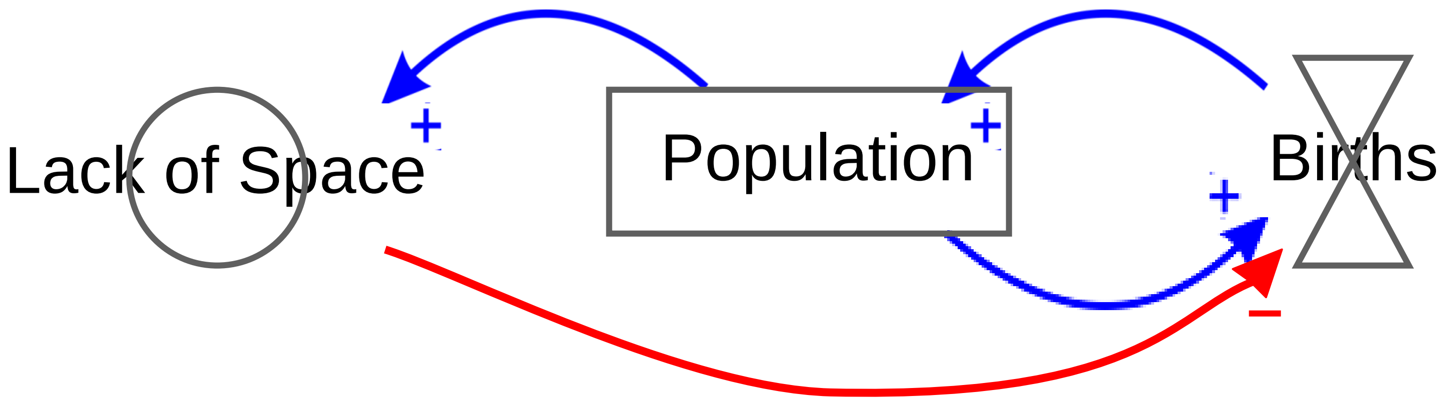
Everything that is predefined for the whole simulation – usually it is a constant but can be a function too.

E.g.: Average Temperature, Number of Cash Desks (In a supermarket), Birth Rate, Maximum capacity of a Room, etc.

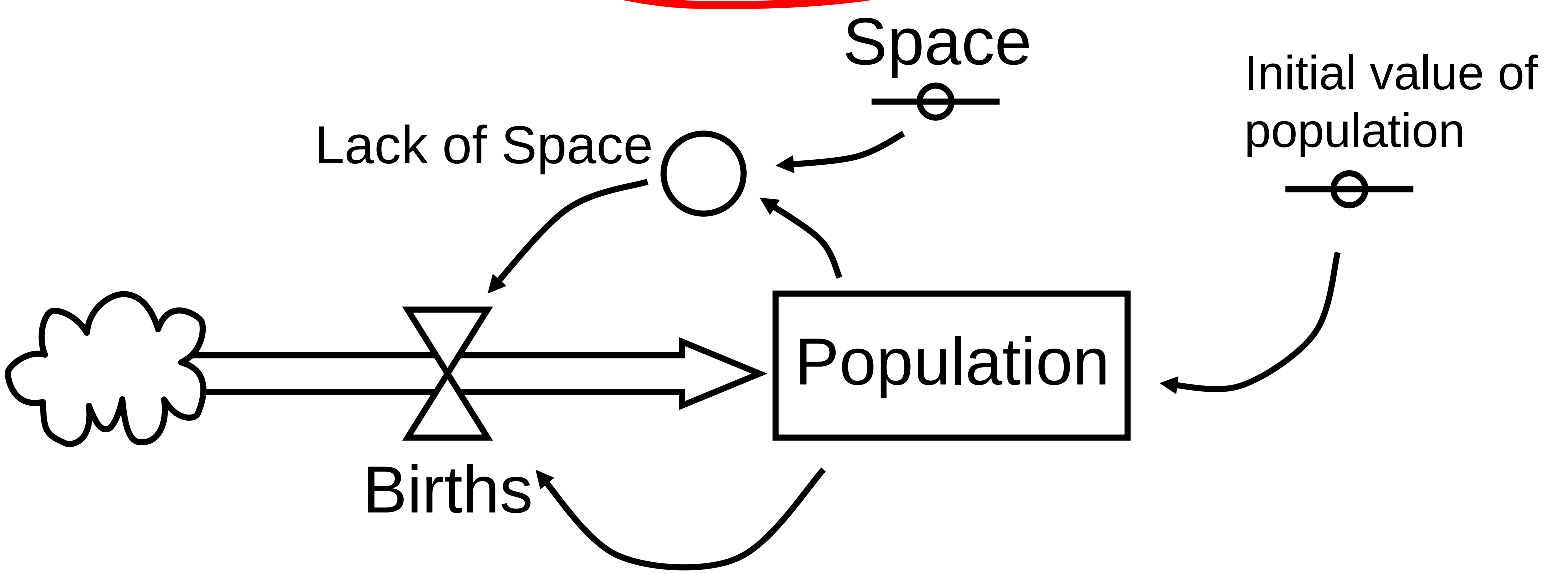


Initial value of state
Source
Change of State
State
Feedback
Rate of change


Population
+
+
Births
Initial value of
population
Births
Population



Lack of Space
+
Population
+
Births
+
−



Space
Initial value of
population
Lack of Space
Population
Births

Quantification?

$$lack\_of\_Space = Space - Population$$?
$$lack_{of_{Space}} = Space - 3 * Population$$?
$$lack\_of\_Space = \frac{Space}{Population}$$?
$$lack\_of\_Space = \frac{Population}{Space}$$?

$$Births = 3 * Population + lack\_of\_Space$$?
$$Births = 10 * Population - lack\_of\_Space$$?
$$Births = 0.2 * Population + \frac{1}{lack\_of\_Space}$$?

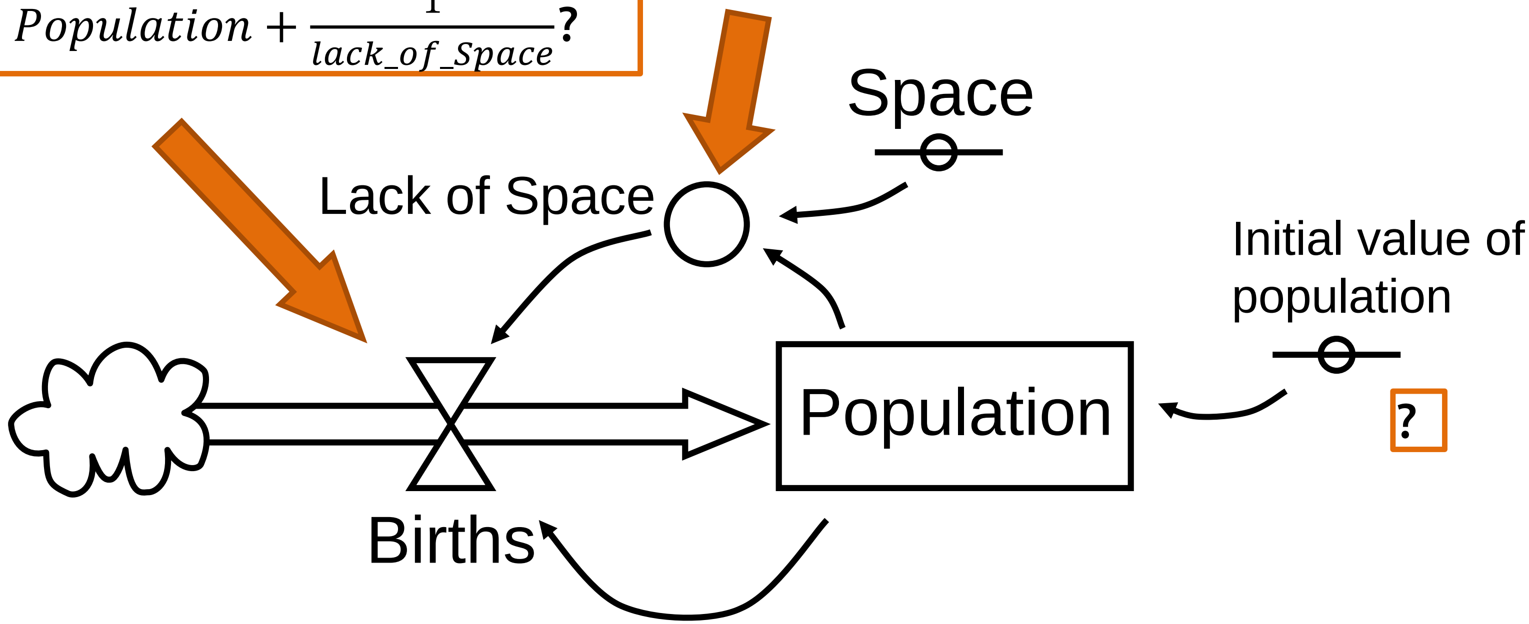


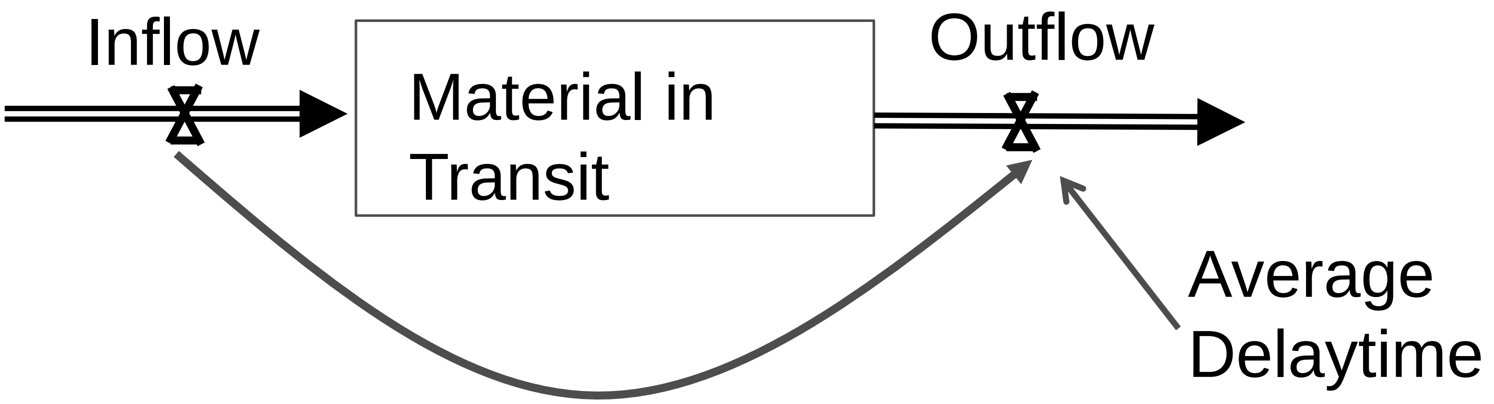
Table Function

- Responsible for nonlinar relationships
- Uses pairs of numbers
  - Interpolation inbetween:
    linear, step, spline, approximation
  - Out of range:
    error, repeat, extrapolate

Delays

The Value of the input will be time-delayed for the delay time:

Output = Material in Transit / Delaytime



Analysis of SD Models

Mathematical Modelling and Simulation

- **Analytical**: Evaluation of equilibrium, behaviour and stability in an area (ordinary differential equations) But: For large systems this can be difficult and not useful for time variant values
- **Base Run**:
  The Model runs with the predefined set of parameters (which represent the best information available at this time).

Simple Structure

Stock and Flow with two flows

Differential Equation:

$$\dot{Stock}(t) = Inflow(t) - Outflow(t)$$

Integral equation*:*

$$Stock = \int_{t0}^{t} \big(Inflow(s) - Outflow(s)\big)\, ds + stock(t0)$$

Equilibrium

Static Equilibrium*:*
Inflow and *Outflow* are 0;
State of the system remains unchanged.

Dynamic Equilibrium*:*
Inflow and *Outflow* are the same;
State of the system remains unchanged

Analysis of SD Models

Mathematical Modelling and Simulation

- **Optimization / Calibration**:
  With specific algorithms some – unknown – parameter values can be calculated by macthing a objective function.
- **Parametervariation / Sensitivity Analysis**:
  Multiple simultion runs are simulated with different sets of parameter values, which are gained from
  - even distributed intervals or
  - stocastically from a probability function

Simulators

Mathematical Modelling and Simulation

- SD-simulators at least offer the most important elements (Flows, Levels, Auxiliaries, Table-functions, etc.) to be preimplemented.
- Additionally parametervariation and optimization is possible with most SD simulators.
- Examples: AnyLogic (does not only support SD), Vensim, Stella, PowerSim...

Conclusion

Mathematical Modelling and Simulation

- System-Dynamics is a top-down modelling approach. Its graphical representation is broadly standartized.
- Important Elements: causal relationships, causal loops, stock and flow diagrams
- It is equivalent to a DE model. Thus results can be analysed using the same methods.
- Simulators: AnyLogic, Vensim, Stella, PowerSim...

Mathematical Modelling and Simulation

Thank you for your attention!

Questions?

Discrete Event Simulation and Modelling with Event Graphs

General

Modelling Approach/
Representation Form

Model Type

Event Graphs $\xrightarrow[\text{to}]{\text{leads}}$ Discrete Event Simulation Model

General

Mathematical Modelling and Simulation

Modelling Approach/ Representation Form		Model Type
Event Graphs	leads to →	Discrete Event Simulation Model

Compare:

System Dynamics or Lagrange Formalism	leads to →	Differential Equation Model

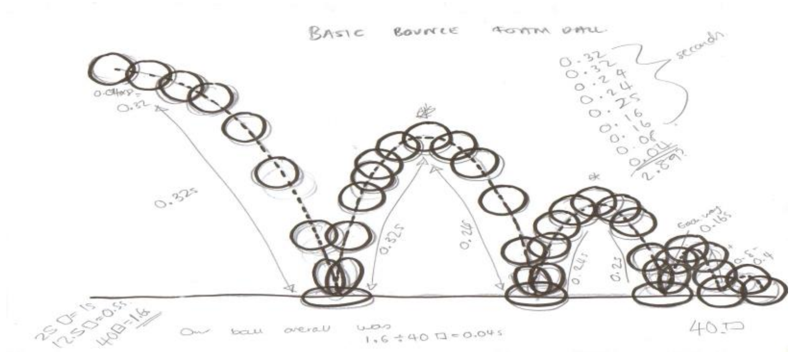
Motivation

Mathematical Modelling and Simulation

- Simulation of systems that change their states only at so called „events“



- (Simulation of systems that can be approximated as such)




Discrete Event Simulation
Fundamental Concept

Mathematical Modelling and Simulation

Two fundamental components of a discrete event simulation (DES) model

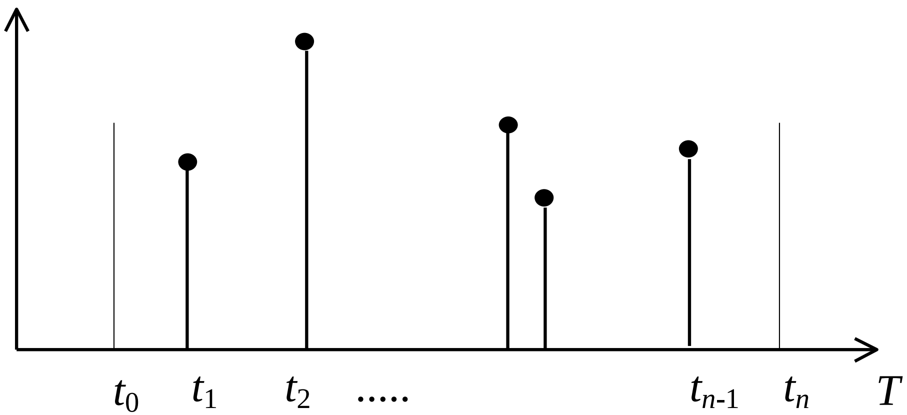
State Variables

„Observables“ or the model. Used to generate the simulation output

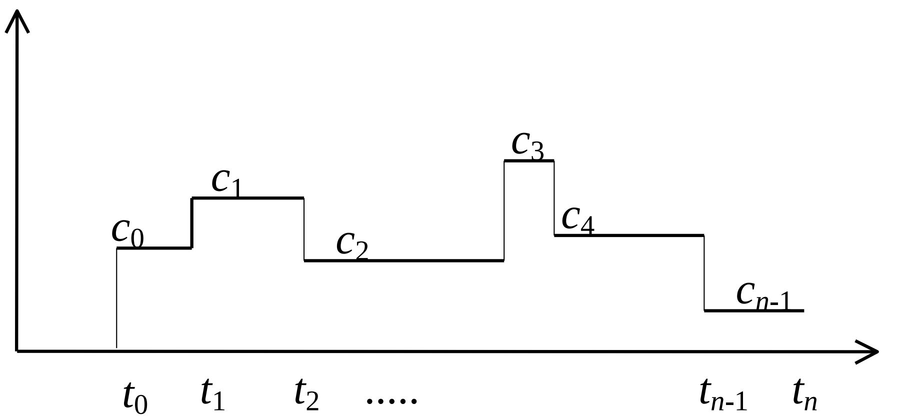
Events:

Cause state variables to change and schedule/cancle future events

- Events



- States piecewise constant



Events are scheduled using

Event Notices.

Every event notice contains two pieces of information:

- What (type of) event is being scheduled, and
- the (simulated) time at which the event is planned to occur

The

Event List

keeps the event notices in order by ranking them based on the lowest scheduled time.

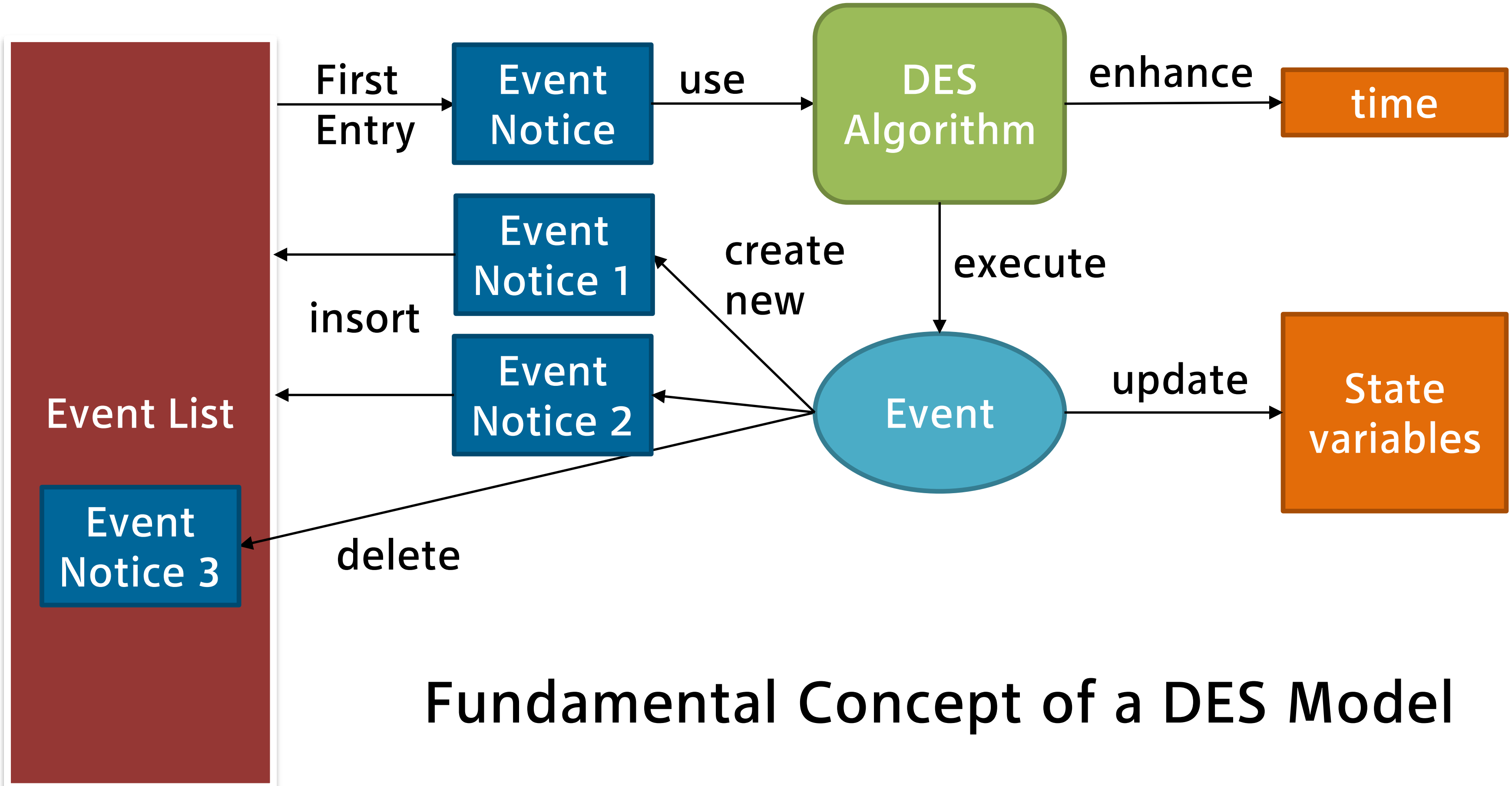
The events list is managed by basic

Discrete Event Algorithm

that controls the flow of time in the simulated world of the model

Discrete Event Simulation
Fundamental Concept

Mathematical Modelling and Simulation



Fundamental Concept of a DES Model

How to formalise DES Models

EVENT GRAPHS

Event Graphs General

- Concept introduced by Lee Schruben in 1983
- Sometimes called „Simulation Graphs“
- Graphical representation of a DES model which can directly be fed to Event Graphs simulators, e.g. SIGMA (Compare with System Dynamics and AnyLogic)
- Very general – for most applications, more specialised concepts / simulators are used

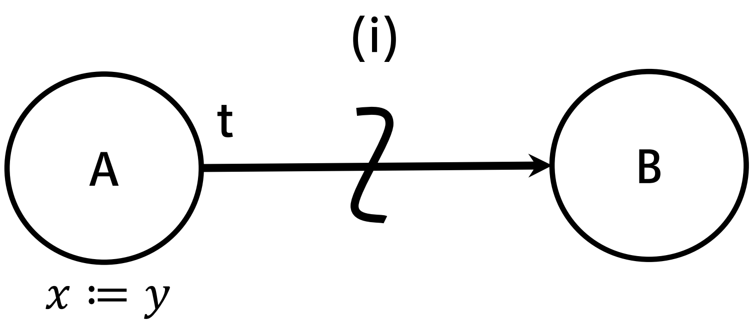
Event Graph Formalism

The occurrence of an event with type A

- causes state variable x to change its state to y

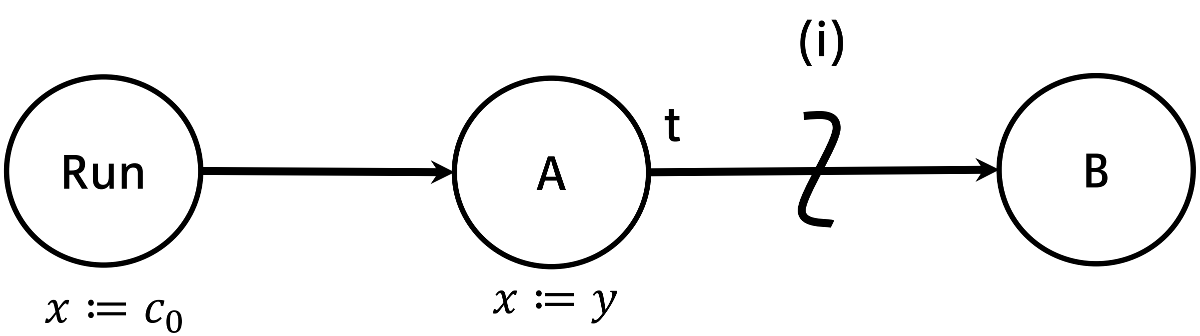
causes an event with type B

- to be scheduled after a time delay of t,
- providing condition (i) is true, after the state transitions for Event A have been performed



Event Graph Formalism

- As the event-list is empty at the beginning of the simulation, a designated initial event needs to be given.
- Usually this event is labelled with „Run“



Example: Difference Equation

- Goal: model the sequence

$$y(k+1) = ay(k) + b,$$
$$k = 0, \dots, t_{end}, \qquad y(0) = y_0$$

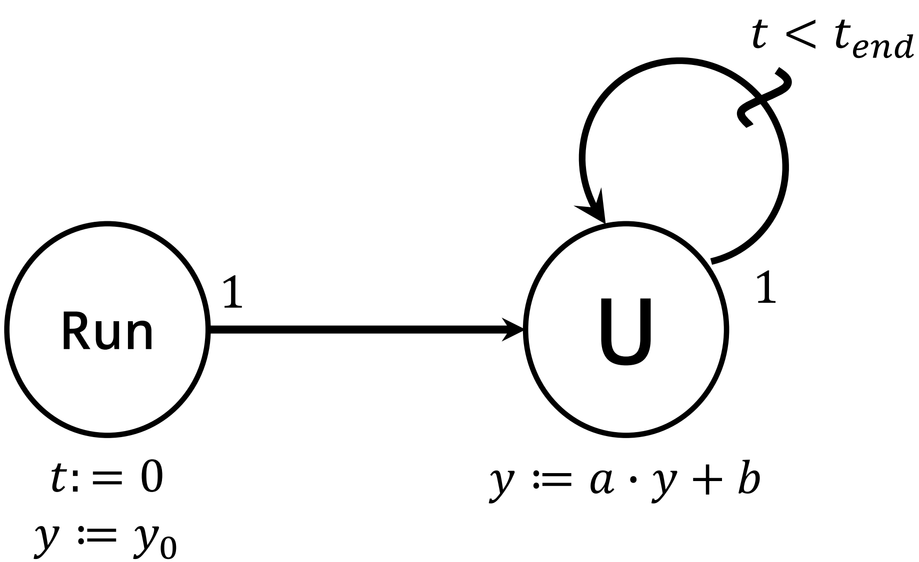
using the Event Graph formalism

Example: Difference Equation

- Goal: model the sequence

$$y(k+1) = ay(k) + b,$$
$$k = 0, \dots, t_{end}, \qquad y(0) = y_0$$

using the Event Graph formalism



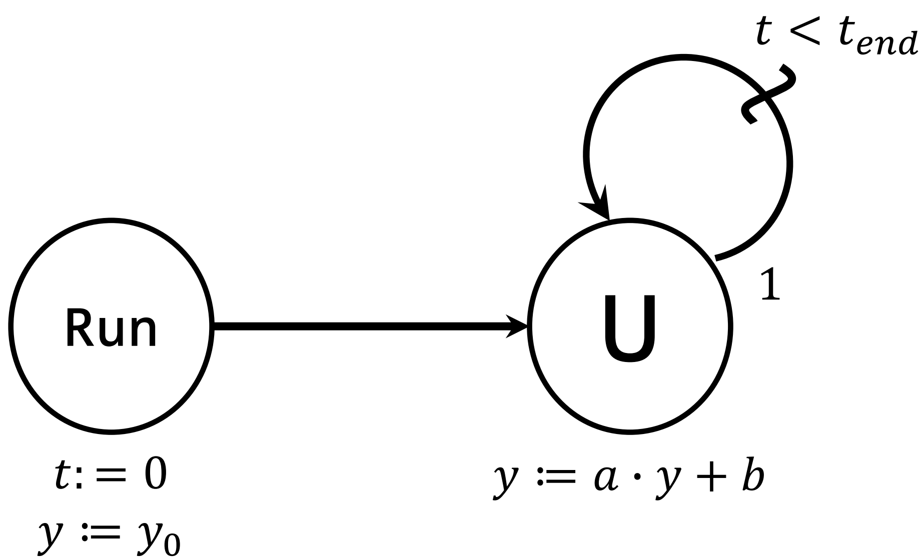
Example: Difference Equation

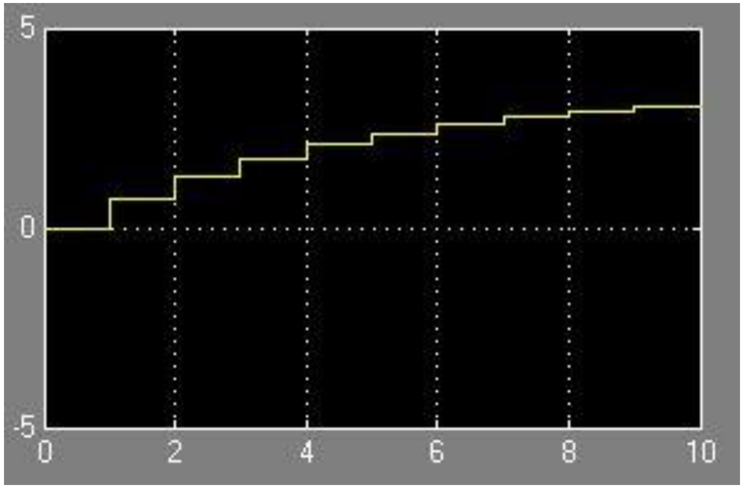
Mathematical Modelling and Simulation

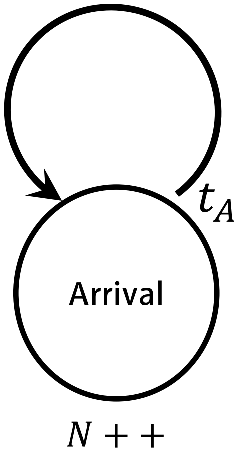
- Goal: model the sequence

$$y(k+1) = ay(k) + b,$$
$$k = 0, \ldots, t_{end}, \qquad y(0) = y_0$$

using the Event Graph formalism





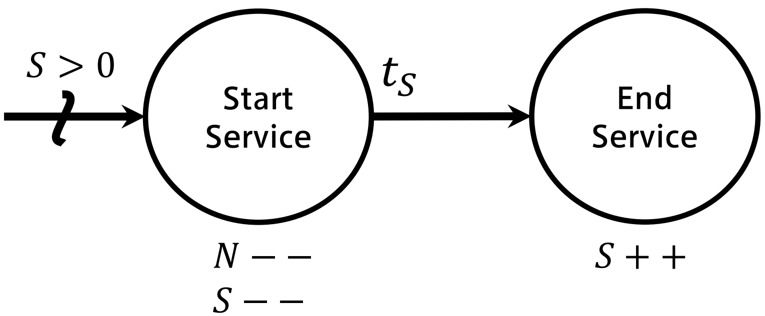


Arrival Process:

- Used to generate „entities" coming from outside the system boundaries
- Usually changes increases a cumulative state variable by one. This variable is usually called a queue
- Sequence of interarrival times t_A that can be
  - constant, a
  - deterministic sequence, or a
  - sequence of random variables

Classical Elements

TU WIEN Mathematical Modelling and Simulation



Service Process:

- Used to treat „entities“ coming from, e.g. an arrival process
- If available ($S > 0$), takes an element from the queue
- Sequence of service times t_S that can be
  - constant, a
  - deterministic sequence, or a
  - sequence of random variables

Multiple Server Queue

- Customers arrive to a service facility according to an arrival process and are served by one of k servers.
- Customers arriving to find all servers busy wait in a single queue and are served in order of their arrival.
- Parameters:

  t_A = interarrival times
  t_s = service times
  k = total number of servers
- State Variables:

  $Q :=$ # of customers in queue
  $S =$ # of available servers

Multiple Server Queue

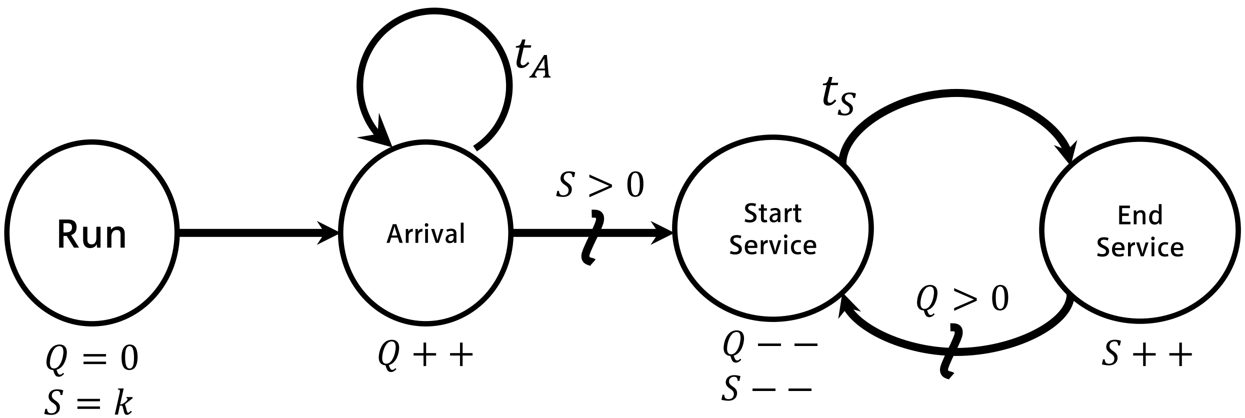
- Customers arrive to a service facility according to an arrival process and are served by one of k servers order of their arrival.

- Parameters:

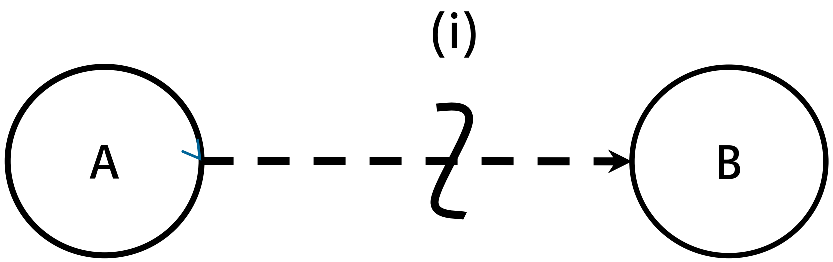
  t_A = interarrival times
  t_s = service times
  k = total number of servers

- State Variables:

  Q := # of customers in queue
  S = # of available servers



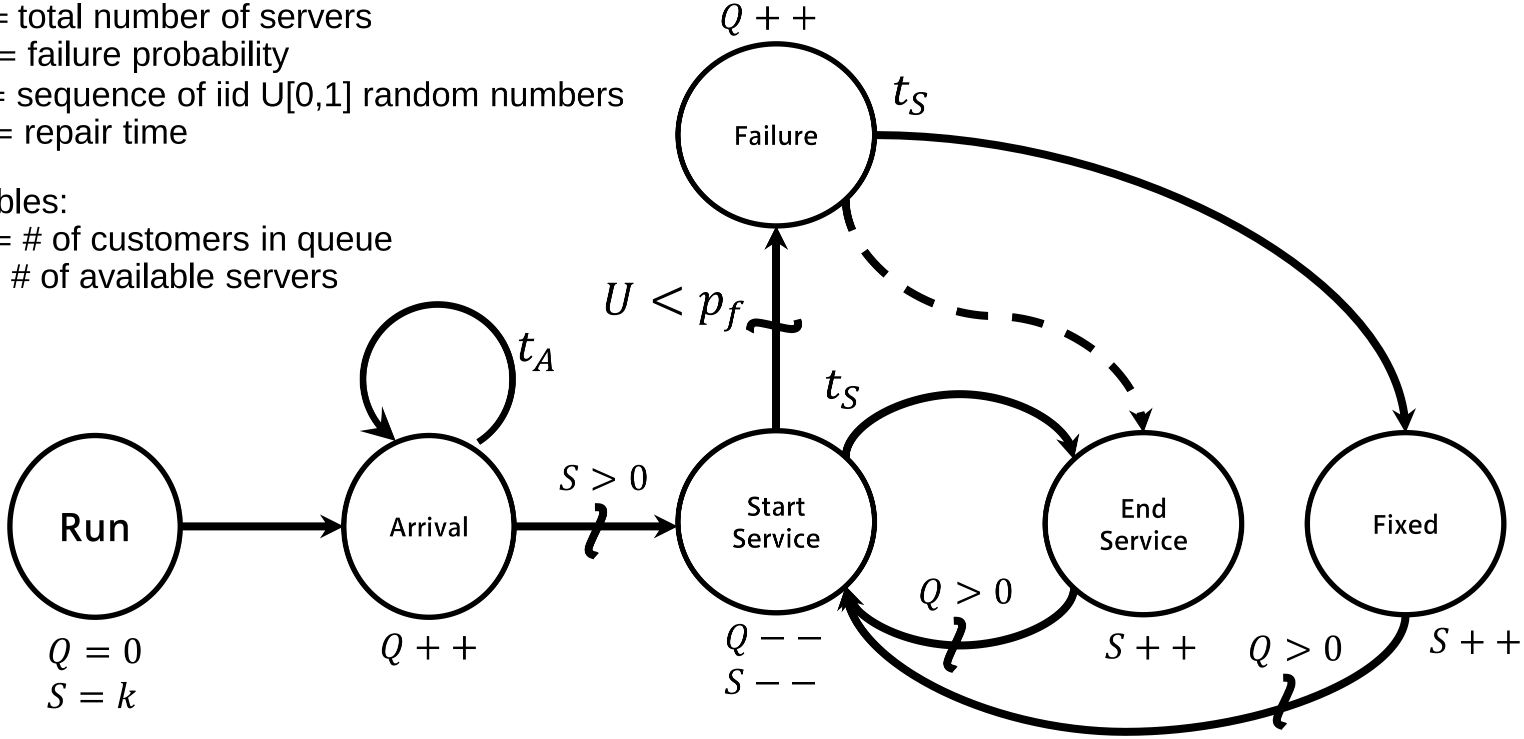
Cancelling Edge



- the inverse operation of the scheduling edge
- whenever event with tyoe A occurs, then if condition (i) is true, the first occurrence of an event with type B is removed from the event list
- if event B is not scheduled to occur, then nothing happens.
- if there are multiple occurrences, only the first is removed.

Multiple Server Queue with Failure

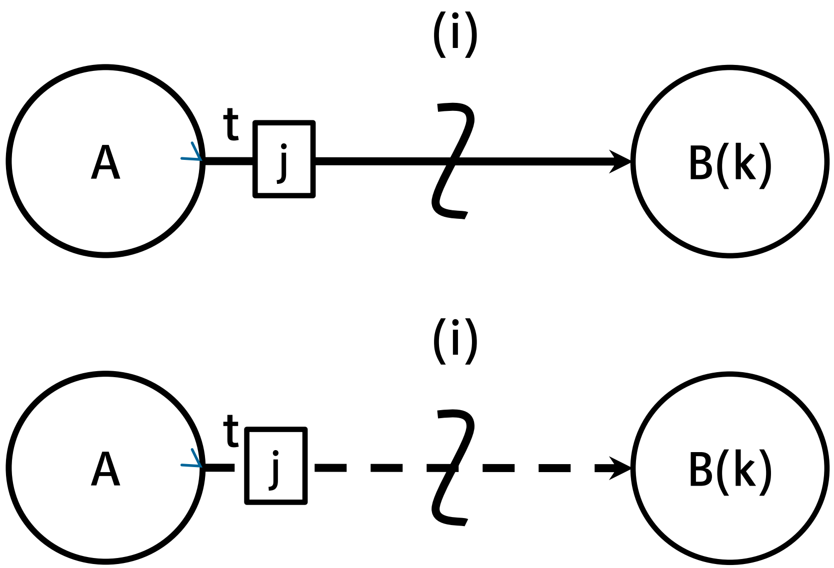
- Customers arrive to a service facility according to an arrival process and are served by one of k servers order of their arrival.
- With certain failure probability the server breaks while serving
- Parameters:
  - t_A = interarrival times
  - t_s = service times
  - k = total number of servers
  - p_f = failure probability
  - U = sequence of iid U[0,1] random numbers
  - t_R = repair time
- State Variables:
  - Q := # of customers in queue
  - S = # of available servers



Parameterization of Events

Mathematical Modelling and Simulation

Scheduling edge with parameter: When A occurs then, if (i) is true, B is scheduled after t time units. When B occurs, its parameter k will be set to the value given by the expression j (j is calculated when A occurs).



Tandem Server Queue

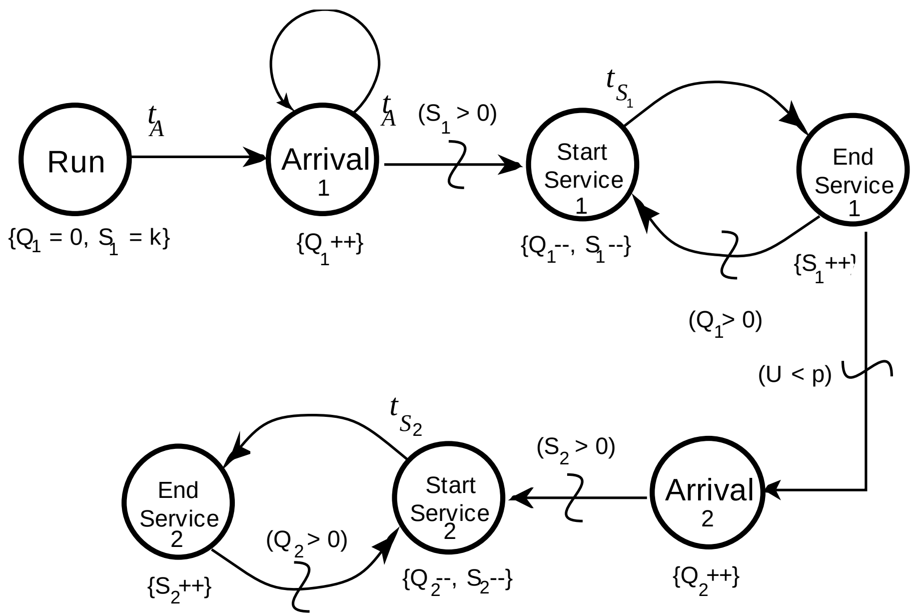
- Customers processed by one workstation consisting of a multiple-server queue.
- Upon completion of service at the first workstation, a customer proceeds with probability p to a second workstation or departs the system with probability (1- p)..
- Parameters:

  t_{A_i} = interarrival times at WS i
  t_{s_i} = service times at WS i
  k_i = total number of servers at WS i
  p = probability to proceed from 1 to 2
  U = sequence of iid U(0,1) random numbers
- State Variables:

  Q_i ≔ # of customers in queue at WS i
  S_i = # of available servers at WS i

Tandem Server Queue

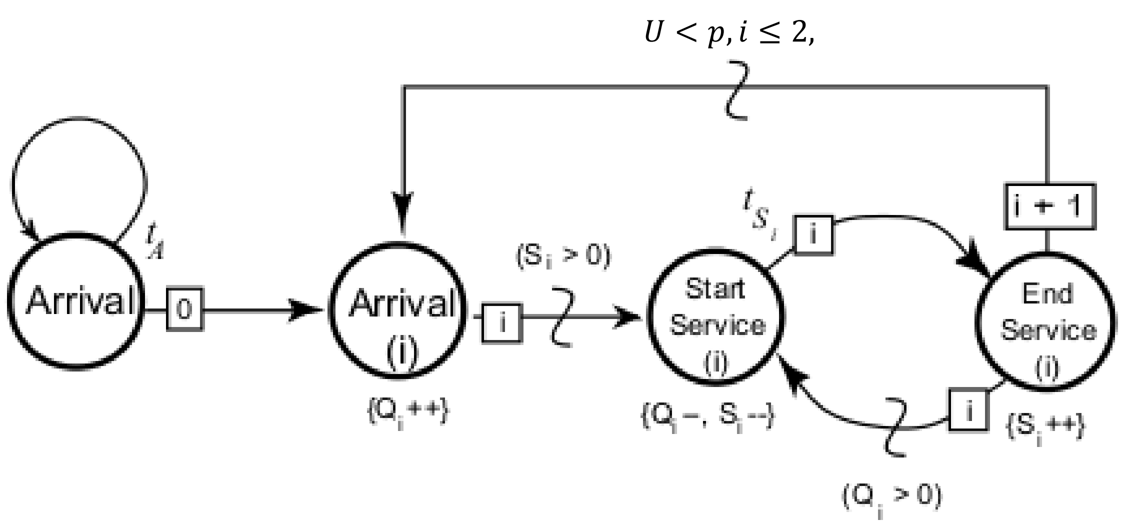
- Customers processed by one workstation consisting of a multiple-server queue.
- Upon completion of service at the first workstation, a customer proceeds with probability p to a second workstation or departs the system with probability (1- p)..



Tandem Server Queue

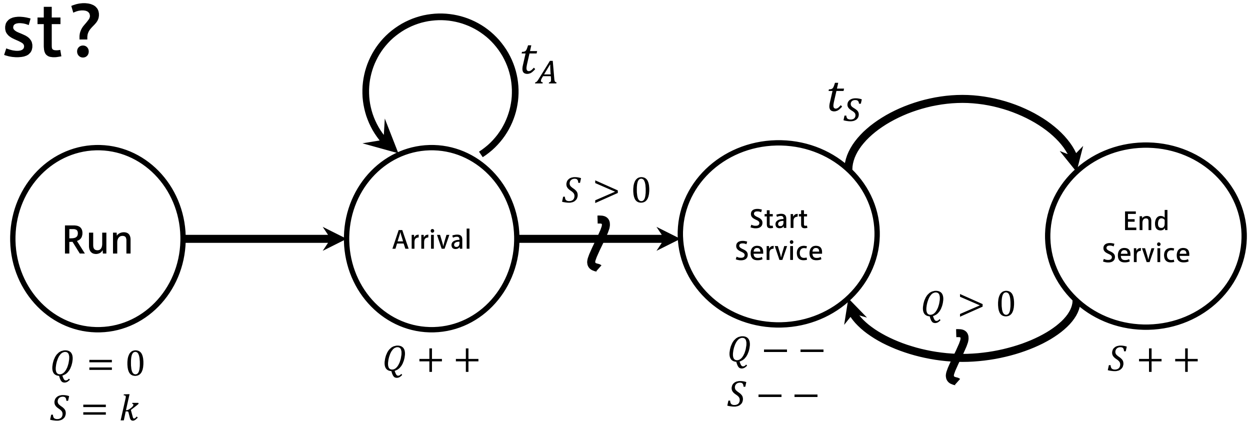
TU WIEN Mathematical Modelling and Simulation

- Customers processed by one workstation consisting of a multiple-server queue.
- Upon completion of service at the first workstation, a customer proceeds with probability p to a second workstation or departs the system with probability (1- p)..



Case Study:

- What happens, when executing a Multiple Server Queue model with deterministic service and arrival times?
- Event Notices?
- Event List?



Event Notices and Parameters

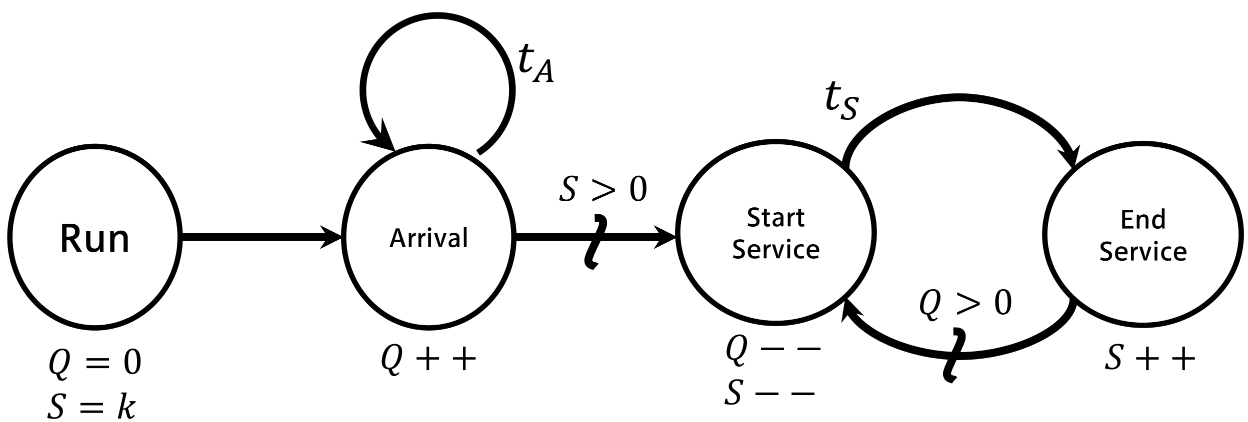
Mathematical Modelling and Simulation

```
DISCRETE start
 server = 2;  queue = 0
 SCHEDULE arrival .AT. t+0.
END ! of start

DISCRETE arrival
 queue = queue + 1;  t_arrival = 1
 SCHEDULE arrival .AT. t+tarr
 IF server .GE. 0 SCHEDULE start_service at t+0.
END ! of arrival
```

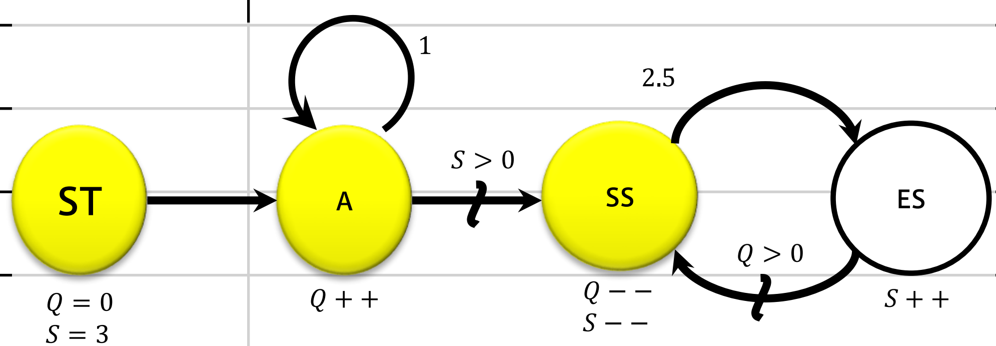
```
DISCRETE start_service
 queue  = queue – 1;  server = server -1
 t_service = 2.5
 SCHEDULE end_service .AT. t+t_service
END ! of start_service

DISCRETE end_service
 server = server + 1
 IF queue .GE. 0 SCHEDULE start_service at t+0.
END ! of end_service
```



Event List Multiple Server Queue

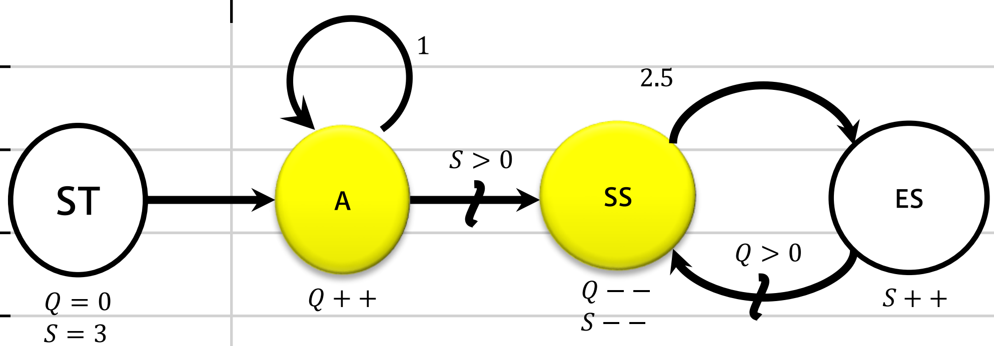
time	event	action	schedule
0	ST	Q=0; S=2;	A at t+0=0
0	A	Q=Q+1=1	A at t+1=1; SS at t+0=0
0	SS	Q=Q-1=0; S=S-1=1	ES at t+2.5=2.5
1	A		
2.5	ES		



Event List Multiple Server Queue

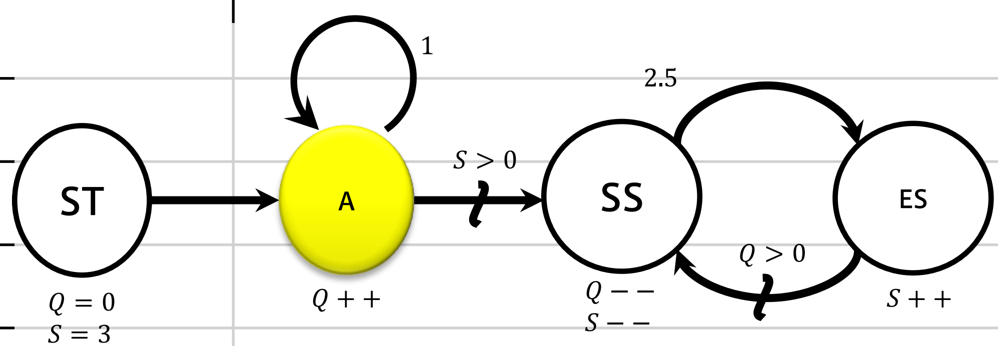
Mathematical Modelling and Simulation

time	event	action	schedule
0	ST	Q=0; S=2;	A at t+0
0	A	Q=Q+1=1	A at t+1=1; SS at t+0=0
0	SS	Q=Q-1=0; S=S-1=1	ES at t+2.5=2.5
1	A	Q=Q+1=1	A at t+1=2; SS at t+0=1
1	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=3.5
2.5	ES		
2	A		
3.5	ES		



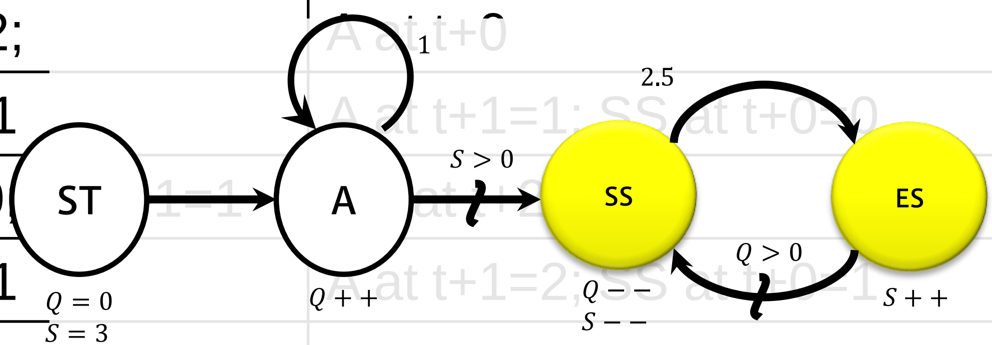
Event List Multiple Server Queue

time	event	action	schedule
0	ST	Q=0; S=2;	A at t+0
0	A	Q=Q+1=1	A at t+1=1; SS at t+0=0
0	SS	Q=Q-1=0; S=S-1=1	ES at t+2.5=2.5
1	A	Q=Q+1=1	A at t+1=2; SS at t+0=1
1	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=3.5
2	A	Q=Q+1=1	A at t+1=3; (SS condition not true)
2.5	ES		
3.5	ES		
3	A		



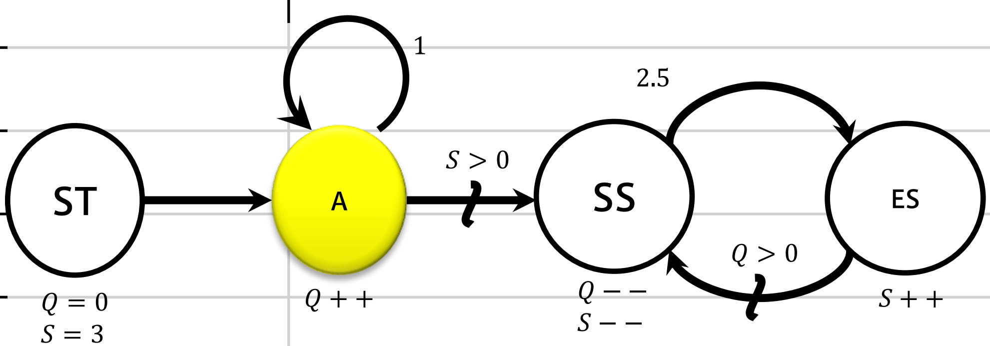
Event List Multiple Server Queue

time	event	action	schedule
0	ST	Q=0; S=2;	A at t+0
0	A	Q=Q+1=1	A at t+1=1; SS at t+0=0
0	SS	Q=Q-1=0;	at t+2
1	A	Q=Q+1=1	A at t+1=2; SS at t+0=1
1	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=3.5
2	A	Q=Q+1=1	A at t+1=3; (SS condition not true)
2.5	ES	S=S+1=1;	SS at t+0=2.5
2.5	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=5
3	A		
3.5	ES		
5	ES		



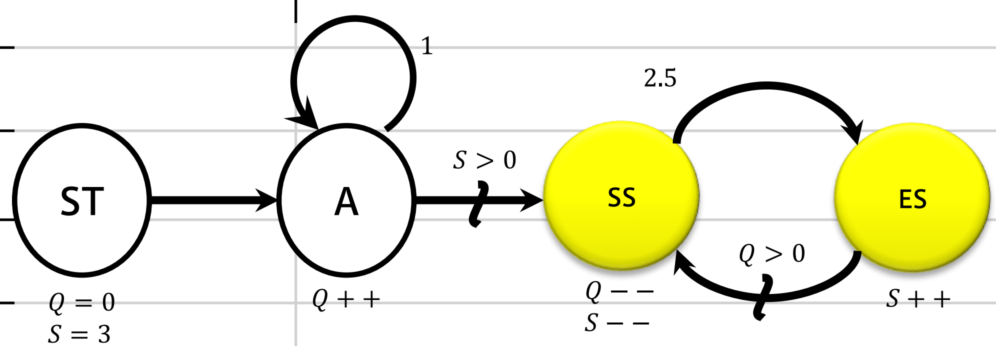
Event List Multiple Server Queue

time	event	action	schedule
2.5	ES	S=S+1=1;	SS at t+0=2.5
2.5	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=5
3	A	Q=Q+1=1	A at t+1=4; (SS condition not true)
3.5	ES		
5	ES		
4	A		



Event List Multiple Server Queue

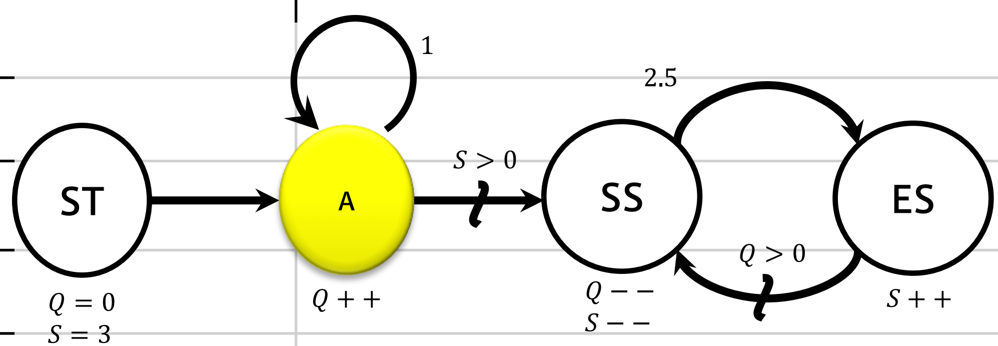
time	event	action	schedule
2.5	ES	S=S+1=1;	SS at t+0=2.5
2.5	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=5
3	A	Q=Q+1=1	A at t+1=4; (SS condition not true)
3.5	ES	S=S+1=1	SS at t+0=3.5
3.5	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=6
4	A		
5	ES		
6	ES		



Event List Multiple Server Queue

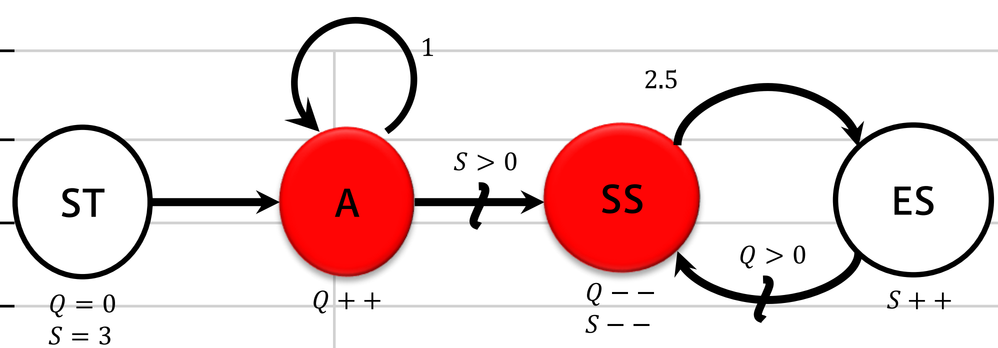
Mathematical Modelling and Simulation

time	event	action	schedule
3.5	ES	S=S+1=1	SS at t+0=3.5
3.5	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=6
4	A	Q=Q+1=1;	A at t+1=5; (SS condition not true)
5	ES		
5	A		
6	ES		



Event List Multiple Server Queue

time	event	action	schedule
3.5	ES	S=S+1=1	SS at t+0=3.5
3.5	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=6
4	A	Q=Q+1=1;	A at t+1=5; (SS condition not true)
5	ES	S=S+1=1;	SS at t+0=5
5	*SS*	simultaneous events – ordering problems	
5	A		
6	ES		



Event List Multiple Server Queue

time	event	action	schedule
3.5	ES	S=S+1=1	SS at t+0=3.5
3.5	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=6
4	A	Q=Q+1=1;	A at t+1=5; (SS condition not true)
5	ES	S=S+1=1;	SS at t+0=5
5	***SS***	Q=Q-1=0; S=S-1=0	ES at t+2.5=7.5
5	A	Q=Q+1=1;	A at t+1=6; (SS condition not true)
6	ES		
7.5	ES		
6	A		

Which one should occur first?
Does it matter?

Event List Multiple Server Queue

time	event	action	schedule
3.5	ES	S=S+1=1	SS at t+0=3.5
3.5	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=6
4	A	Q=Q+1=1;	A at t+1=5; (SS condition not true)
5	ES	S=S+1=1;	SS at t+0=5
5	A	Q=Q+1=2;	A at t+1=6; SS at t+0=5
5	***SS***	Q=Q-1=1; S=S-1=0	ES at t+2.5=7.5
6	ES		
7.5	ES		
6	A		
5	SS		

Which one should occur first?
Does it matter?

Event List Multiple Server Queue

Mathematical Modelling and Simulation

time	event	action	schedule
3.5	ES	S=S+1=1	SS at t+0=3.5
3.5	SS	Q=Q-1=0; S=S-1=0	ES at t+2.5=6
4	A	Q=Q+1=1;	A at t+1=5; (SS condition not true)
5	ES	S=S+1=1;	SS at t+0=5
5	A	Q=Q+1=2;	A at t+1=6; SS at t+0=5
5	SS	Q=Q-1=1; S=S-1=0	ES at t+2.5=7.5
5	SS	Q=Q-1=0; **S=S-1=-1**	ES at t+2.5=7.5
6	ES		
6	A		
7.5	ES		

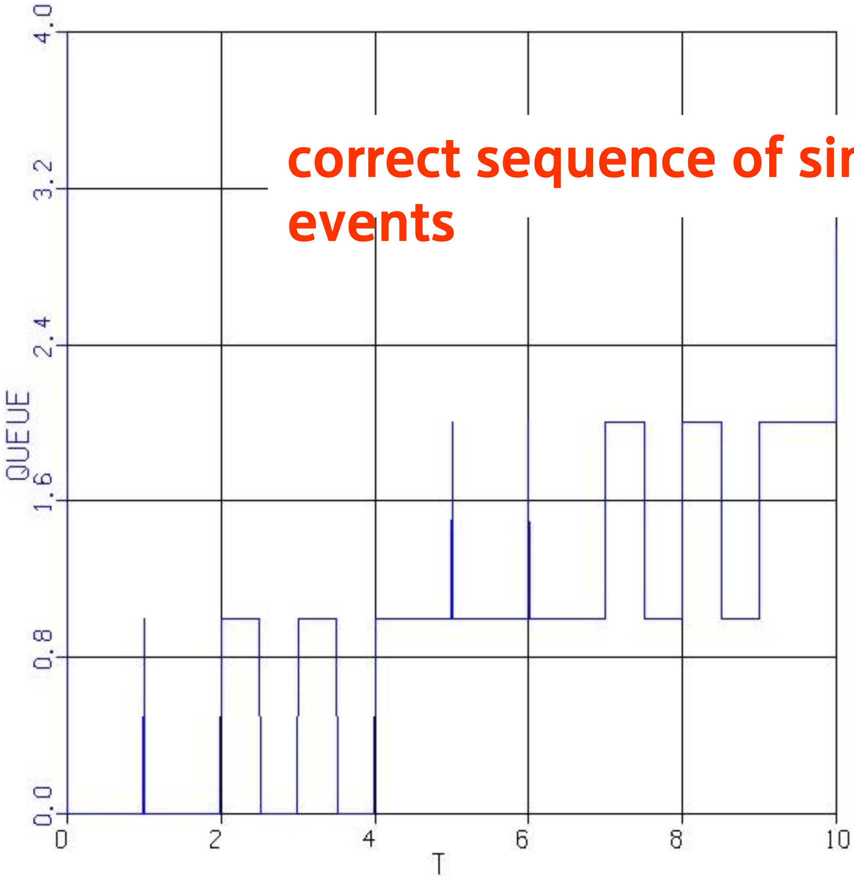
**WRONG ORDER,
WRONG RESULTS**

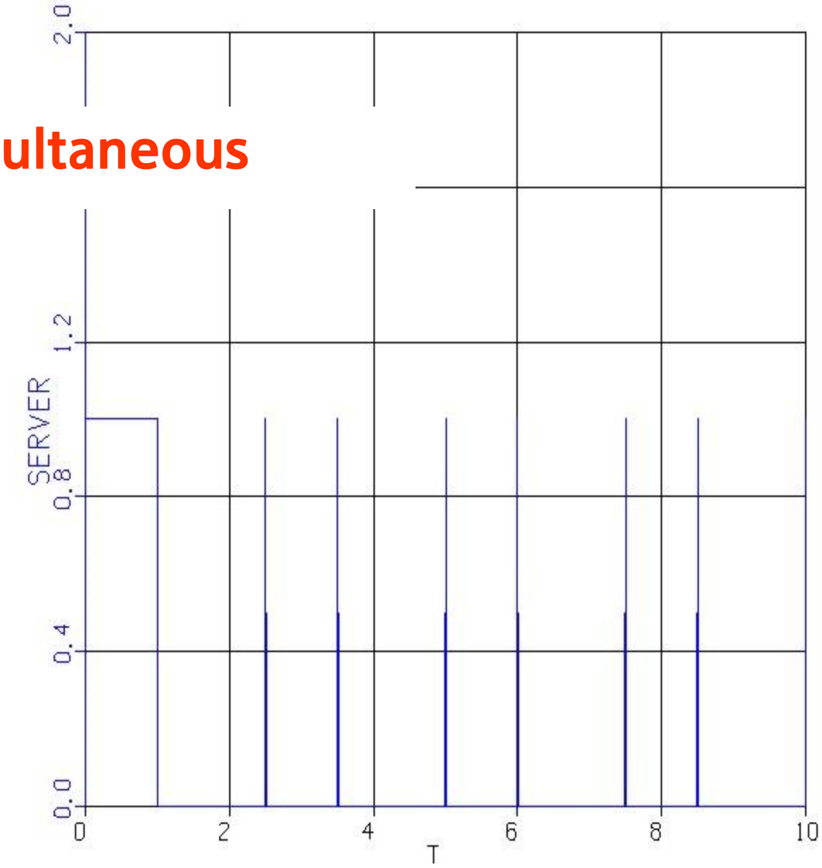
Simultaneous Events

Mathematical Modelling and Simulation

- Simultaneous events occur when more than one event is schedule to occur the exactly the same time.
- In some cases the order of execution of the events is irrelevant, but in other cases certain permutations of the order of occurrence impact the outcome dramatically, often leading to invalid state trajectories and inadmissible values of state variables.
- Event Graph methodology provides the capability of prioritizing scheduling edges, so that simultaneous occurrences of the scheduled event always occur before other scheduled events.
- Although these edge priorities are typically not indicated on the graph itself, all software implementations of Event Graph methodology support edge prioritization.

Simulation Multiple Server Queue

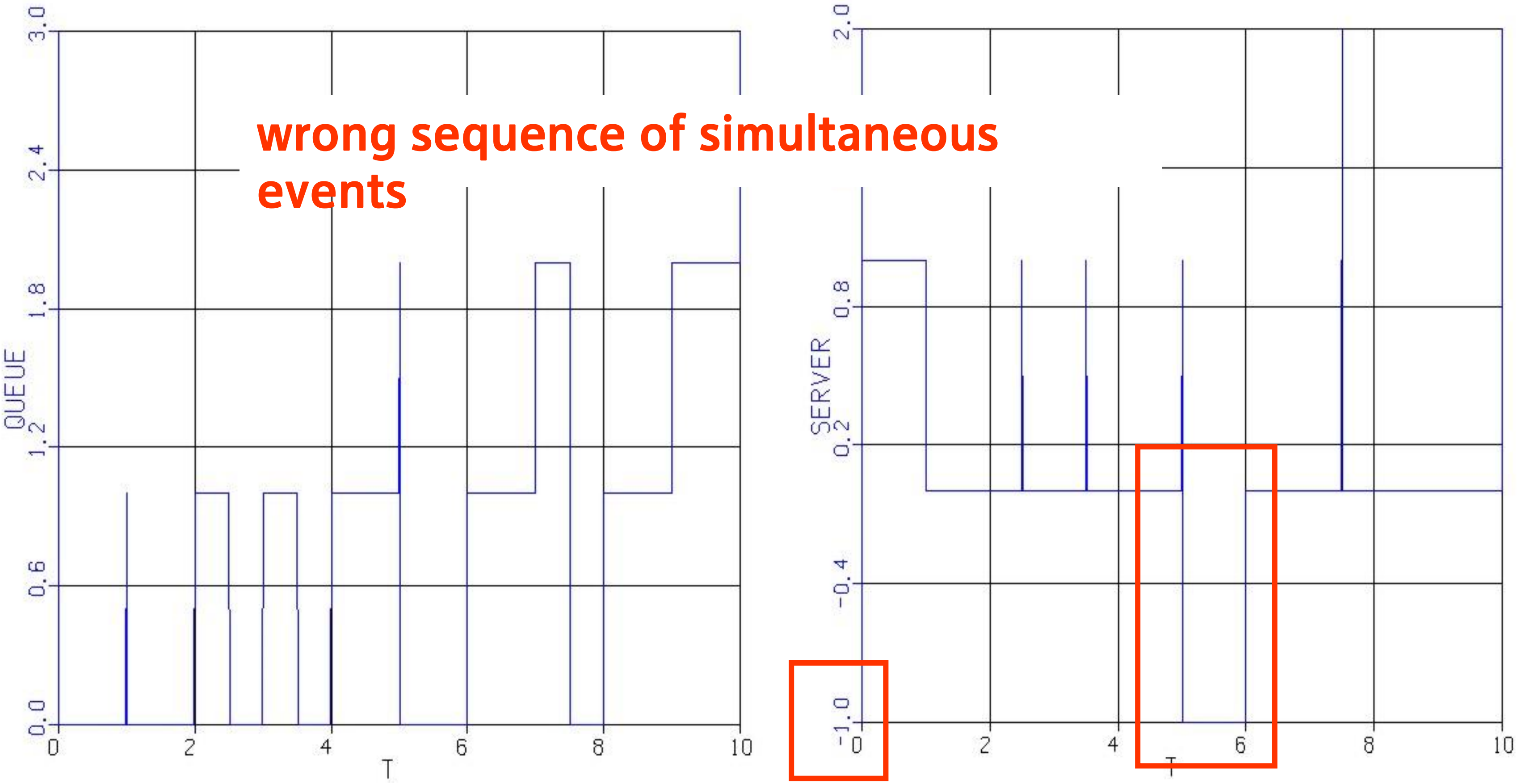




correct sequence of simultaneous events

t_arrival = 1, t_service = 2.5, max_server = 2

Simulation Multiple Server Queue



t_arrival = 1, t_service = 2.5, max_server = 2

ANALYSIS OF QUEUING MODELS

Terminology

- Abbreviation of Queues:

Arrival Time	Service Time	Servers
Determinisitic D	Determinisitic D	One 1
Markovian M	Markovian M	Multiple m
General G	General G	

$\Rightarrow$ Possible combinations:
D/D/1, M/D/m, G/D/m, M/M/m, ...

Terminology

Mathematical Modelling and Simulation

- „Deterministic“: t is Constant
- „Markovian“: Distribution of t is memoryless. I.e. Exponentially distributed $t \sim E(\lambda)$
  $\Rightarrow$ times become a Markov-process
- „General“ : Distribution of t is arbitrary (positive)

- Deterministic Queues (D/D/1, D/D/m):

$$\frac{servicetime}{servers} > arrivaltime$$
$$\Rightarrow unstable$$

$$\frac{servicetime}{servers} \leq arrivaltime$$
$$\Rightarrow stable$$

Analysis of Queues

Mathematical Modelling and Simulation

- Stochastic Queues (M/M/1,G/M/m,...):

$$\frac{E(servicetime)}{servers} \geq E(arrivaltime)$$
$$\Rightarrow unstable$$

$$\frac{E(servicetime)}{servers} < E(arrivaltime)$$
$$\Rightarrow stable$$

Analysis of Queues

Mathematical Modelling and Simulation

- Notation
  - Y_k – time elapsed between (k-1)th and k-th arrival

    $E(Y_k) = \frac{1}{\lambda}$ … average interarrival time
    (λ is the average arrival rate)

  - Z_k – k-th customer service time

    $E(Z_k) = \frac{1}{\mu}$ … average service time
    (μ is the average service rate)

  - W_k – k-th customer waiting time
  - $X(t)$ – average queue length

Analysis of Queues

Mathematical Modelling and Simulation

- Customer system time

  $S_k = W_k + Z_k$, the time k-th customer spends in the system

  $E(W_k) = W$ … average waiting time

  $E(S_k) = T$ … average system time, $T = W + \frac{1}{\mu}$
- Little's law
  - $\bar{N}$ … average number of customers in the system

$$\bar{N} = \lambda T$$

  - special cases

    $\overline{X(t)} = \bar{N}_q = \lambda W$ … average number of customers in the queue

    $\bar{N}_s = \frac{\lambda}{\mu}$ … average no. of customers in service

Results M/M/1 queues:

- Average waiting time in the queue

$$W = \frac{1}{\mu - \lambda} - \frac{1}{\mu} = \frac{\rho}{(1-\rho)\mu}, \qquad \rho = \frac{\lambda}{\mu}$$

- Average length of the queue

$$\overline{X(t)} = \bar{N}_q = \lambda W = \frac{\rho^2}{1-\rho}$$

- Average system time of customers

$$T = W + \frac{1}{\mu} = \frac{1}{\mu - \lambda} = \frac{1}{(1-\rho)\mu}$$

- Average number of customers in the system

$$\bar{N} = \lambda T = \frac{\rho}{1-\rho}$$

Results M/G/1 queues:

- Exponential distribution of interarrival times
- Service times are mutually independent and distributed arbitrarily with parameters

$E(Z_k) = \frac{1}{\mu}$ in $var(Z_k) = \sigma^2$, we define also $\rho = \frac{\lambda}{\mu}$

- Average queue length

$$\overline{X(t)} = \bar{N}_q = \frac{\rho^2}{2(1-\rho)}(1+\mu^2\sigma^2)$$

- Average number of customers in the system

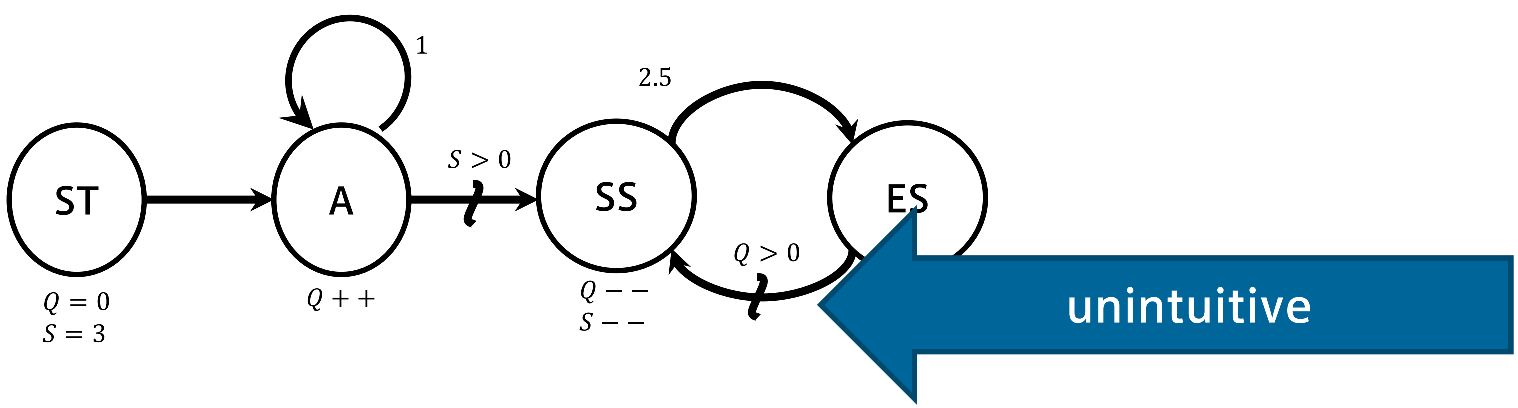
$$\bar{N} = \bar{N}_q + \rho = \frac{\rho}{1-\rho} - \frac{\rho^2}{2(1-\rho)}(1-\mu^2\sigma^2)$$

OTHER SIMULATION ENVIRONMENTS

Other Simulation Environments

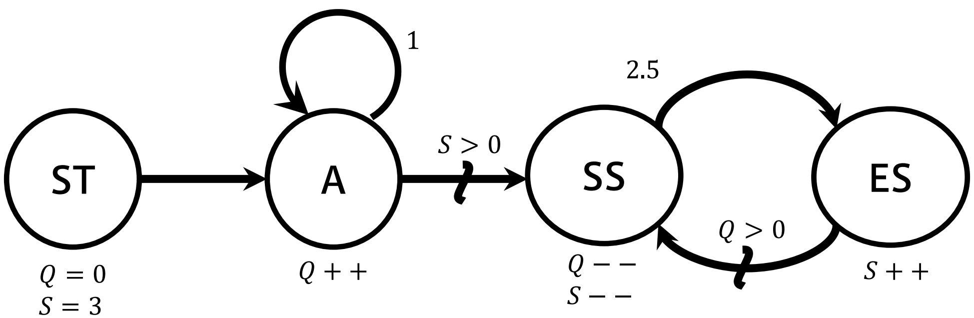
Mathematical Modelling and Simulation

- Most DES models are based on entities being processed in a sysrem
- Therefore they use very similar process structures
- Event Graph description sometimes unnecessary general and unintuitive

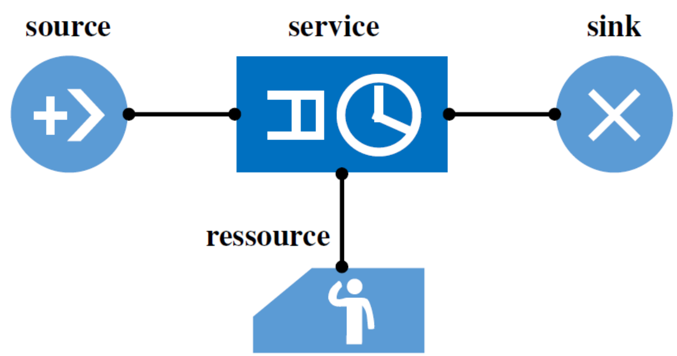


Other Simulation Environments

- DES Simulators for simulation of processes usually use a more intuitive description

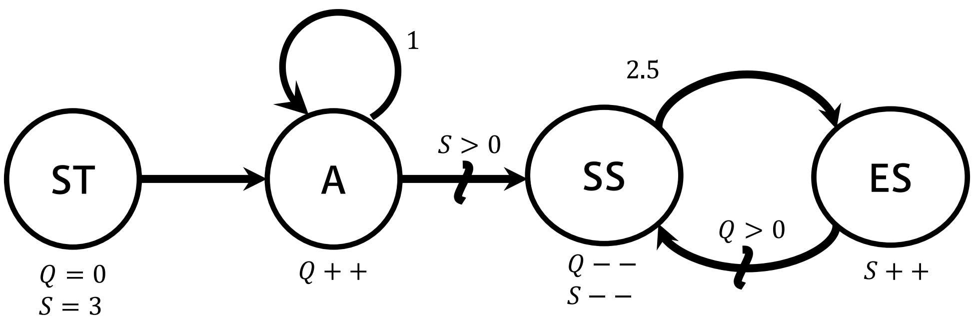


DES Modeling in AnyLogic

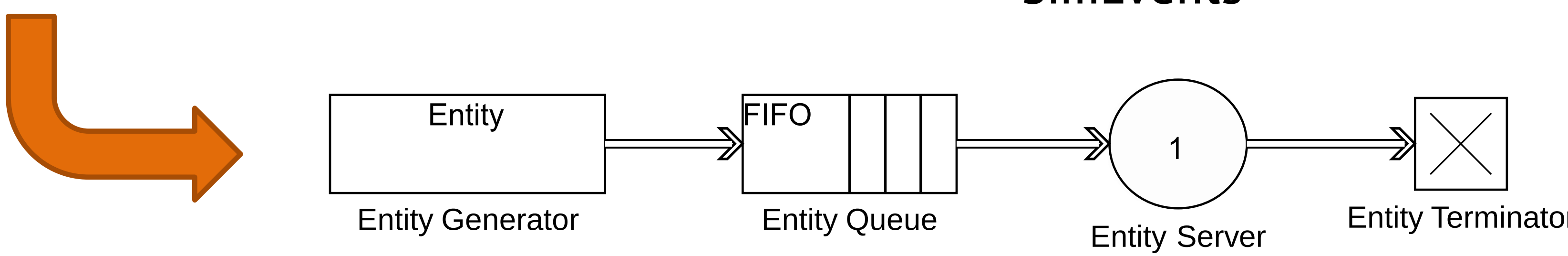


Other Simulation Environments

- DES Simulators for simulation of processes usually use a more intuitive description



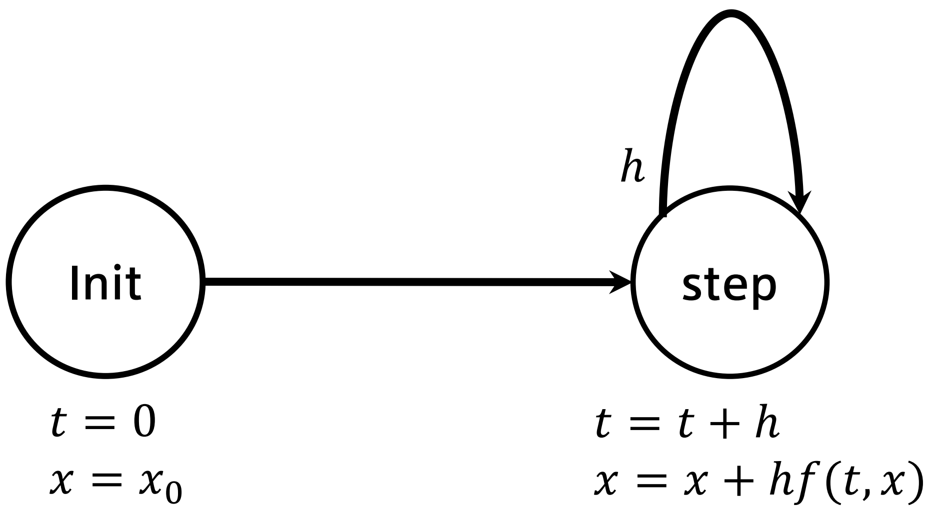
DES Modeling in SimEvents



EVENT GRAPHS BEYOND ENTITIES

Beyond Queues

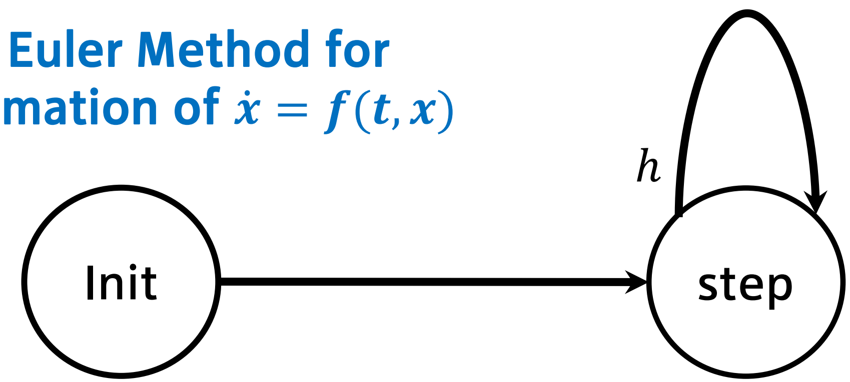
- DES / Event Graphs not only interesting for queuing systems.



Beyond Queues

- DES / Event Graphs not only interesting for queuing systems.

Explicit Euler Method for approximation of $\dot{x} = f(t, x)$



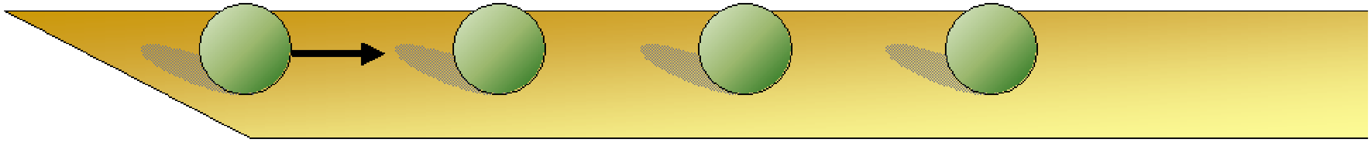
$t = 0$
$x = x_0$

$t = t + h$
$x = x + hf(t, x)$

Beyond Queues

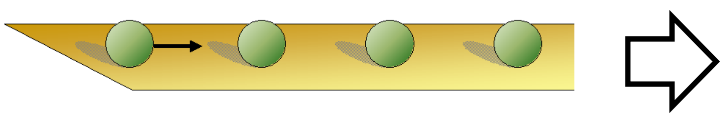
- DES / Event Graphs not only interesting for queuing systems.

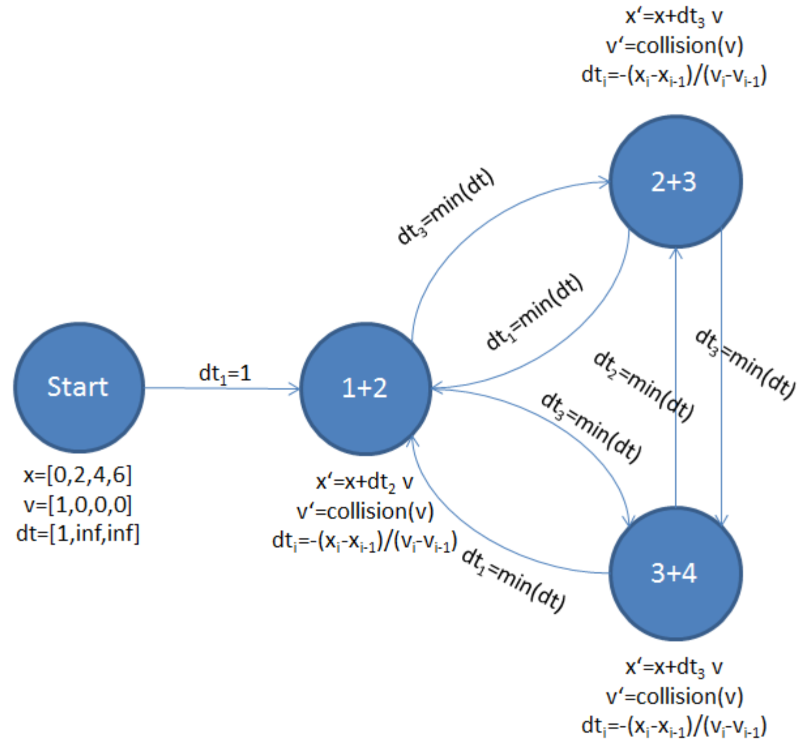
Case Study 1: Collision of Spheres



Beyond Queues

- DES / Event Graphs not only interesting for queuing systems.





Discrete Event and Multi-Method Simulation with Anylogic

Modelling Method vs Model

Mathematical Modelling and Simulation

Modelling Approach/ Representation Form		Model Type
Event Graphs	leads to →	Discrete Event Simulation Model
Compare: System Dynamics	leads to →	Differential Equation Model

Modelling Method vs Model

Mathematical Modelling and Simulation

Modelling Approach/
Representation Form

Model Type

Event Graphs
SimEvents GUI
Anylogic GUI

leads to

Discrete Event Simulation Model

Compare:

System Dynamics
Lagrange Formalism
Modelica/Dymola GUI

leads to

Differential Equation Model

Modelling Method vs Model

Mathematical Modelling and Simulation

Modelling Approach/
Representation Form

Model Type

Event Graphs
SimEvents GUI
Anylogic GUI

leads to

Discrete Event Simulation Model

Compare:

System Dynamics
Lagrange Formalism
Modelica/Dymola GUI

leads to

Differential Equation Model

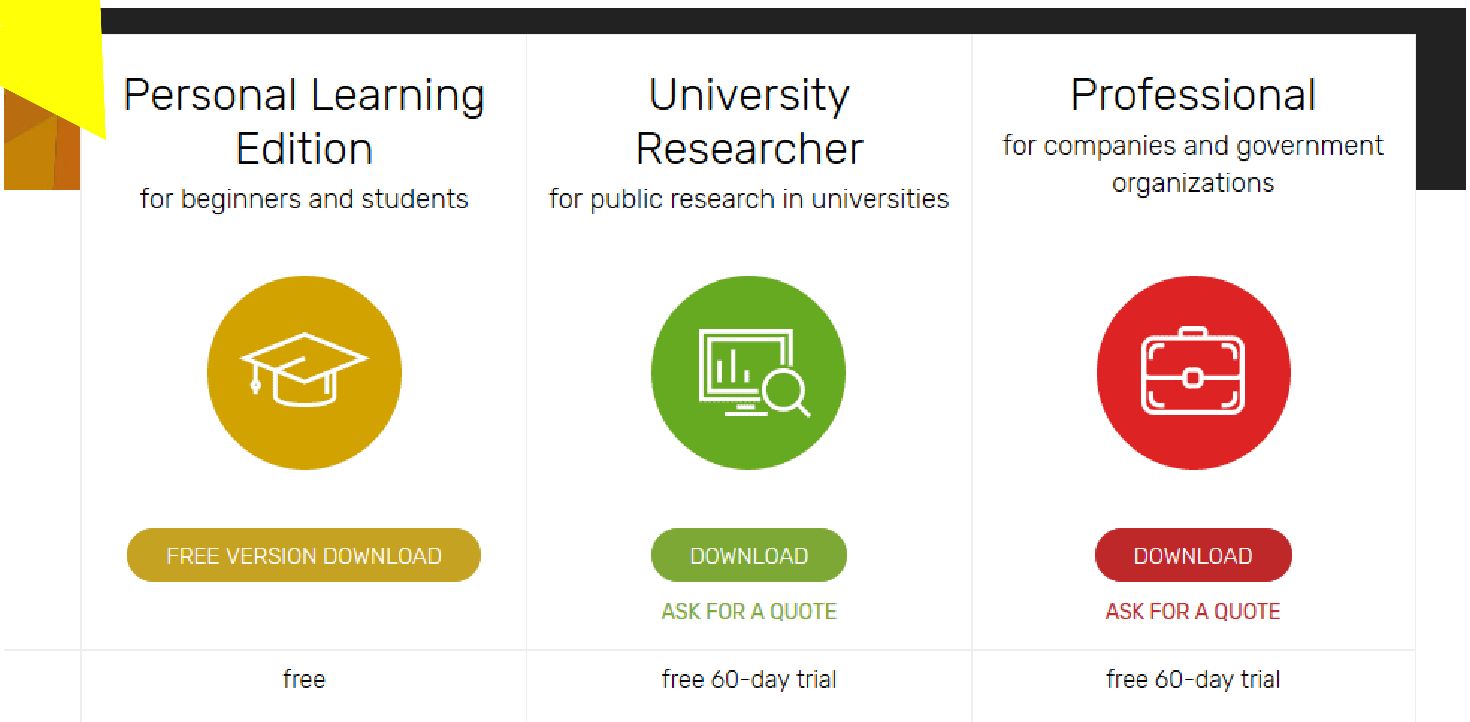
Getting started…

Mathematical Modelling and Simulation

https://www.anylogic.com/downloads/

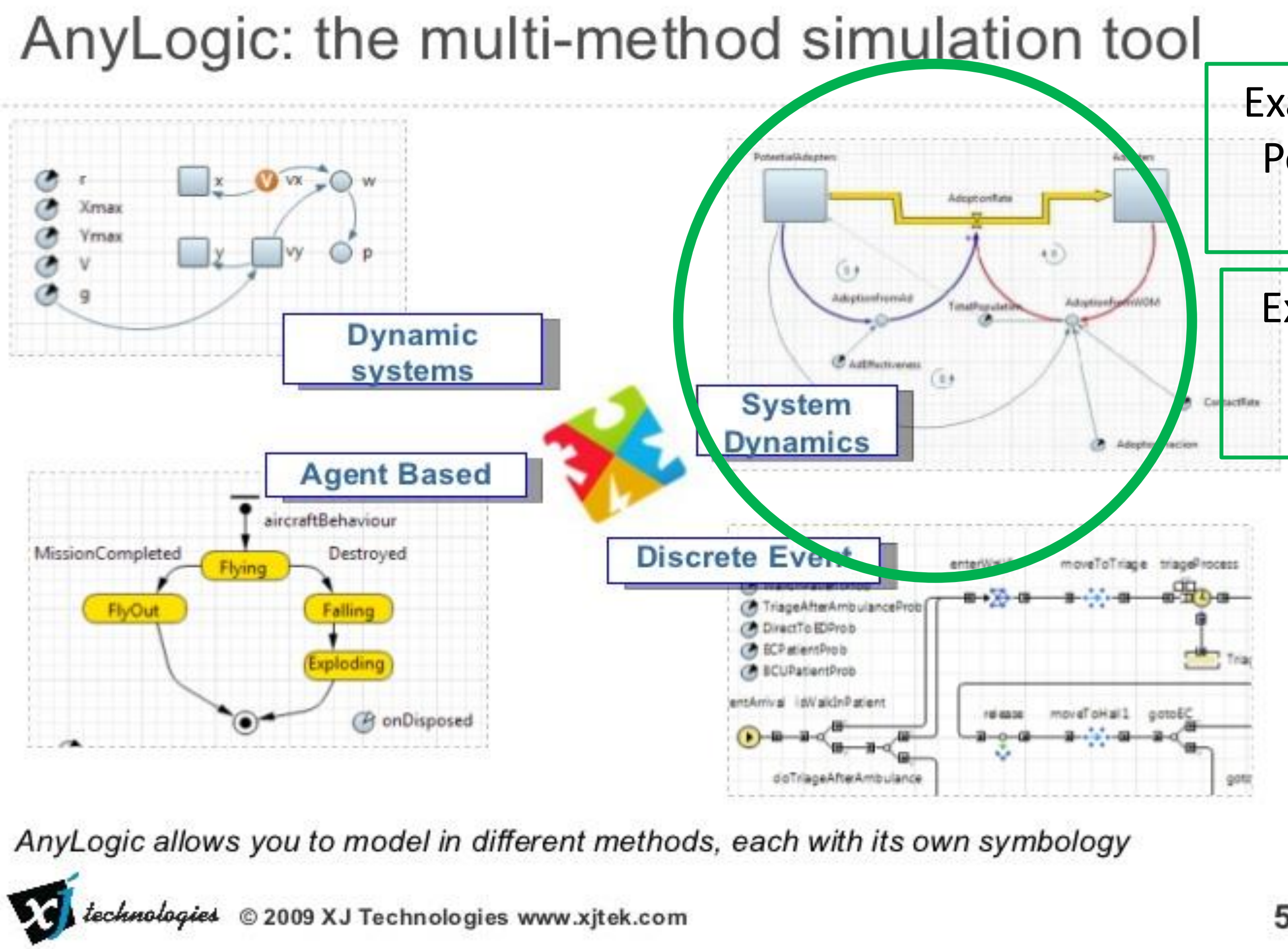
or

USB Stick



What is AnyLogic?

Mathematical Modelling and Simulation

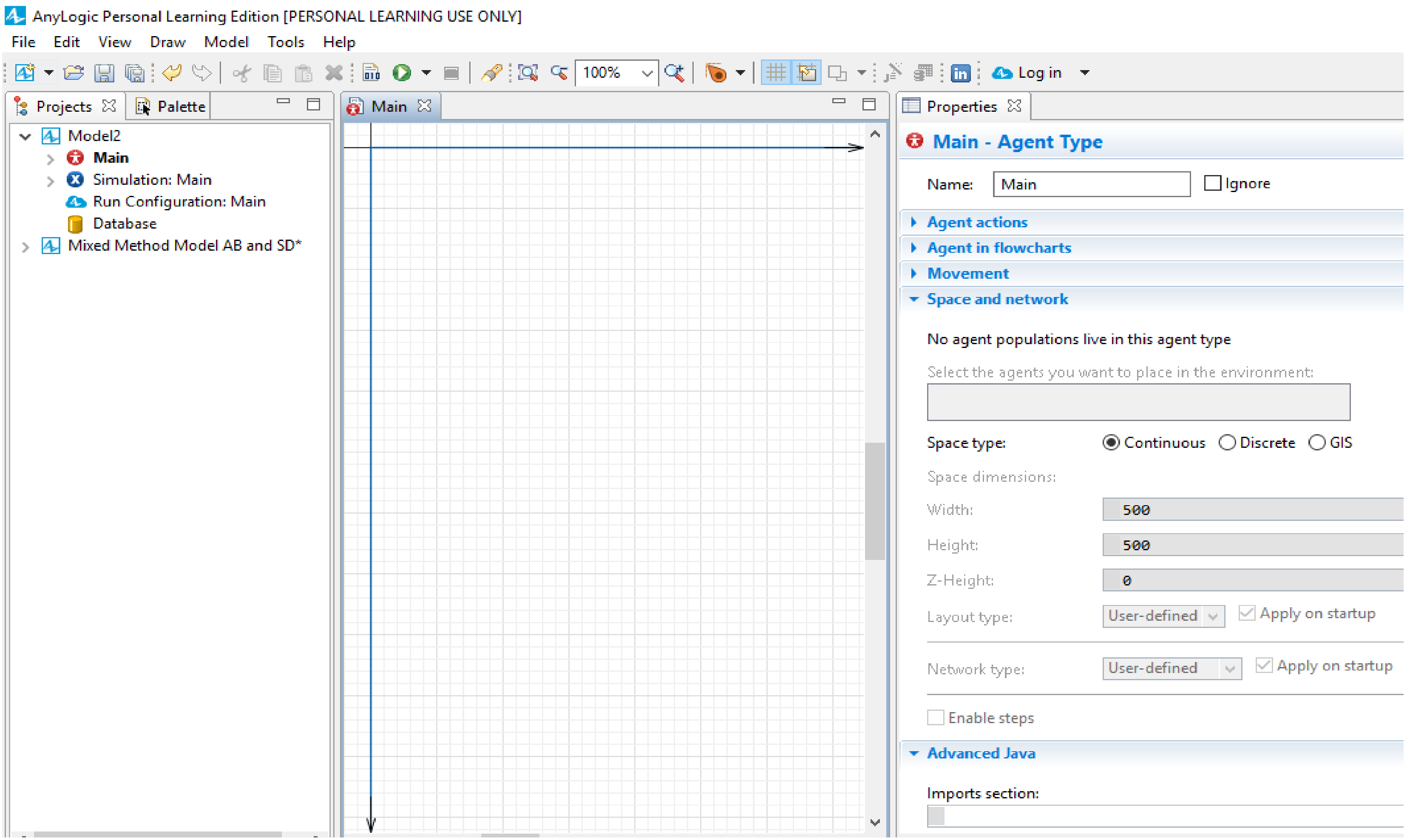


Example 1:
Potential
Sales

Example 2:
Logistic
Model

Basics

Mathematical Modelling and Simulation



- AnyLogic Cloud: run models online from a web browser on any device, including phones and tablets, and share the models with other users.
- https://cloud.anylogic.com/
- Export models to the cloud

EXAMPLE: PREDATOR-PREY MODEL IN ANYLOGIC

General Idea:

The model describes the development of **two populations**.

Population size depends on **births** and **deaths**.

Births depend on the population size.

Predator births also depends on the prey.

The predator population diminishes the prey population.

The predator death rate is independent from prey.

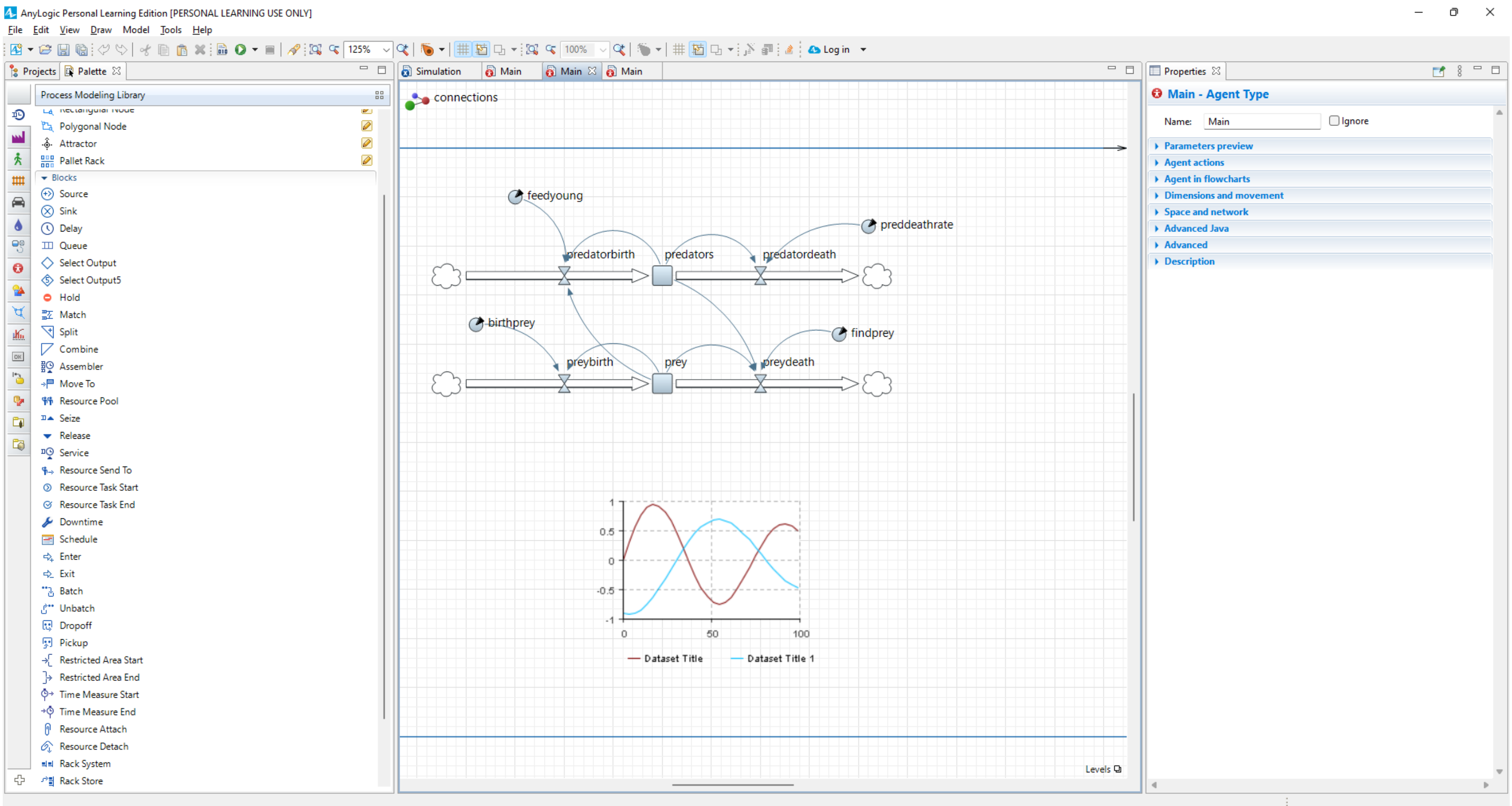
Mathematical Modelling and Simulation

Model Equations:

$$\dot{prey} = (birthprey - findprey * predator) * prey$$
$$\dot{predator} = (feedyoung * prey - preddeathrate) * predator$$

Lets build the model…

Predator-Prey Model



MULTI METHOD MODELLING

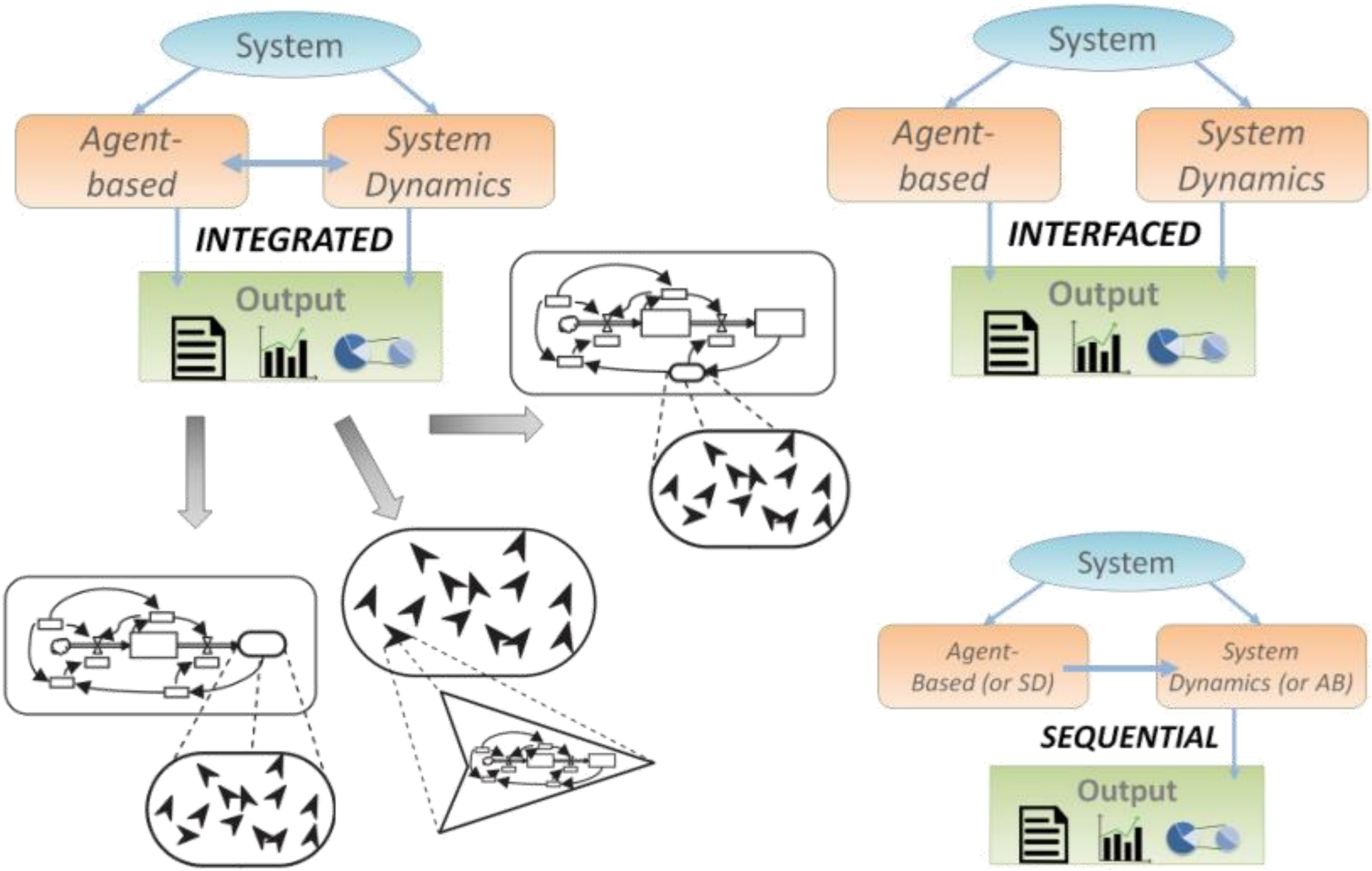
Definition

If a *system can be decomposed into subsystems* and *a model is applied to such a subsystem*, this is called a **submodel.**

A **multi-method model** is *a model that consists of at least two submodels, where at least two different modelling techniques are used.* These submodels *exchange information in some way. This process of information exchange* is called **combining.**

Different Types of Multi Method Models

Mathematical Modelling and Simulation

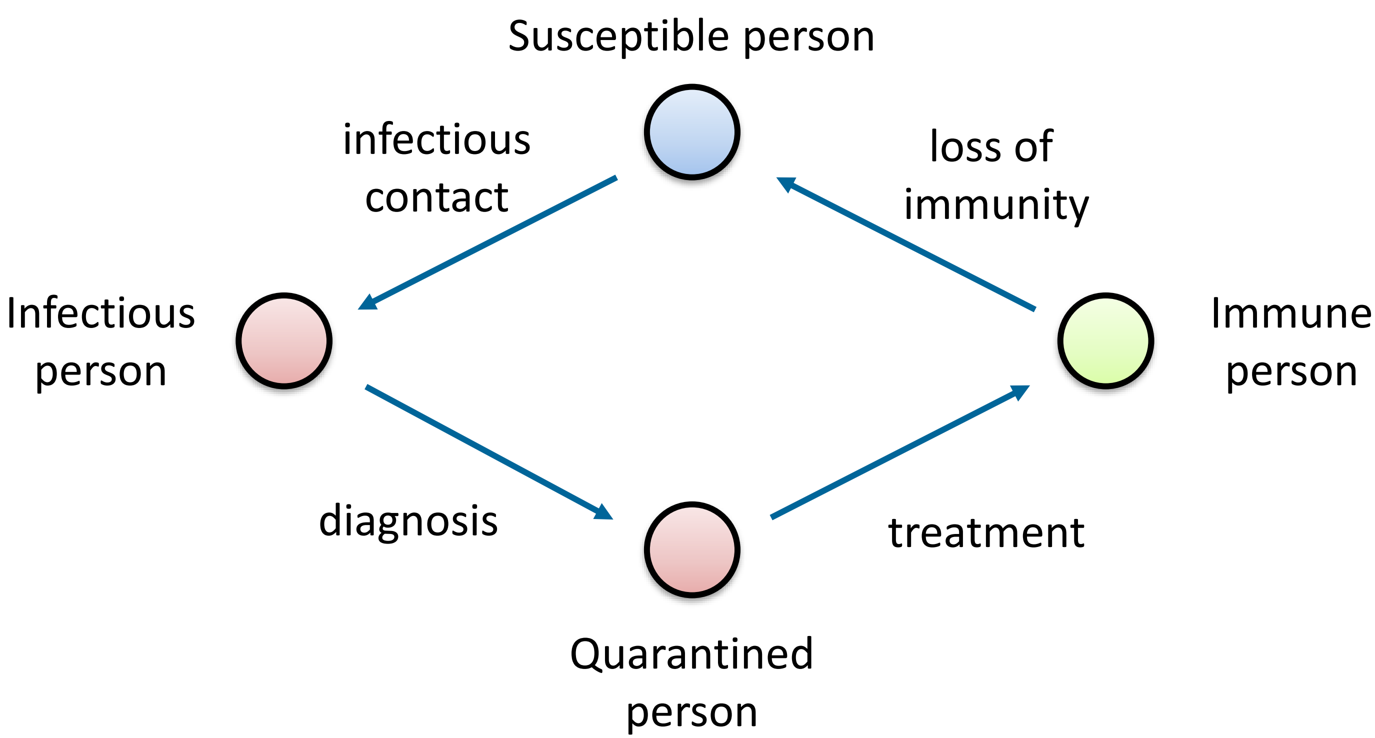


Example: SIRS Epidemic

Mathematical Modelling and Simulation

Research Question:

Investigate the utilization of health-care facilities (e.g. hospitals) in case of the outbreak of an epidemic



Research Question:

Investigate the utilization of health-care facilities (e.g. hospitals) in case of the outbreak of an epidemic

Modelling Problem:

Modelling a disease requires either a nonlinear macroscopic model or a microscopic model with contacts

$\Updownarrow$

Modelling utilization of processes is best modelled with servers and queues.

Example: SIRS Epidemic

Mathematical Modelling and Simulation

Research Question:

Investigate the utilization of health-care facilities (e.g. hospitals) in case of the outbreak of an epidemic

Modelling Problem:

Modelling a dise … copic model or a

System Dynamics

⇕

Modelling … ervers and

Discrete Event Simulation

Example: SIRS Epidemic

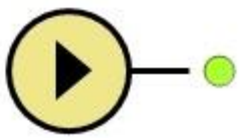
Mathematical Modelling and Simulation

Let‘s build the model….

EXAMPLE: AIRPORT MODEL IN ANYLOGIC

Anylogic GUI: Blocks for DES Models

Mathematical Modelling and Simulation



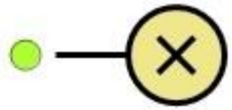
Source

Initializes the event „Arrival *of Entity/Entities“*

Parameters:

-) Arrival Rate & Interarrival time: When do Entities arrive?

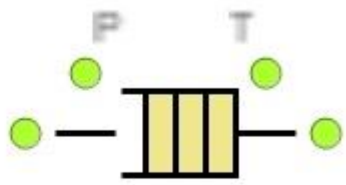
-) Entities per Arrival: How much?



Sink

Initializes event *„Remove Entity/Entities“*

Passive without parameters



Queue

Initializes event *„Waiting Line“*

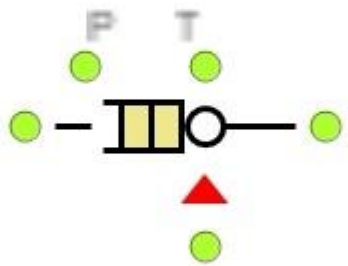
<u>Parameters</u>:

-) Capacity

-) Timeout

-) Preempted abort

Anylogic GUI: Blocks for DES Models



Seize

Initializes event *„get resources“*

<u>Parameters</u>:

-) Number of resources

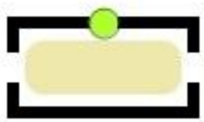
-) Includes a queue

-) Timeout

-) Preempted abort

Stays attached until R*elease*

Anylogic GUI: Blocks for DES Models



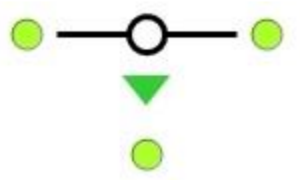
Resource Pool

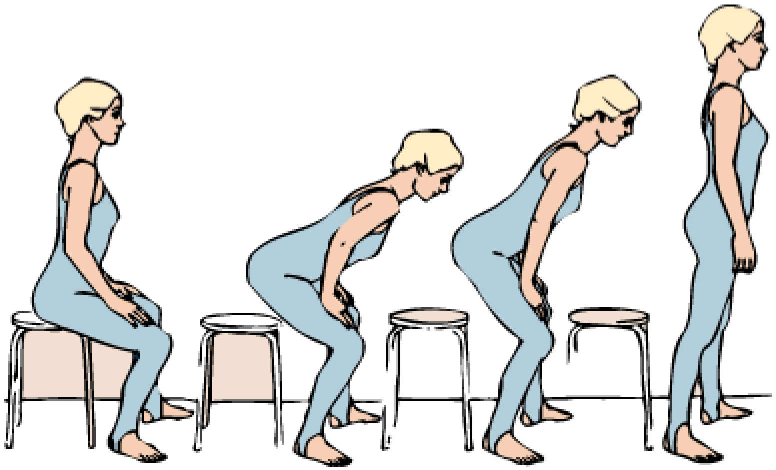
Container of resources of same kind

<u>Parameters</u>:

-) Capacity (absolute or schedule)

-) Is used by *Seize*, *Release* and *Service*





Release

Initializes event *„Release Resource“*

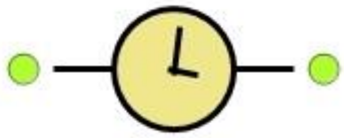
<u>Parameters</u>:

-) Capacity

-) Coupled to a *Resource Pool*

Anylogic GUI: Blocks for DES Models

Mathematical Modelling and Simulation



Delay

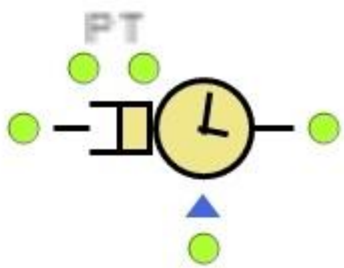
Initializes event *„Wait“*

<u>Parameters</u>:

-) Waitingtime

-) Capacity

Anylogic GUI: Blocks for DES Models



Server

Initializes event „*Processing*“

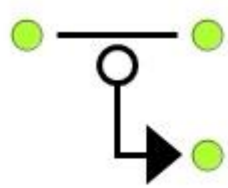
<u>Parameters</u>:

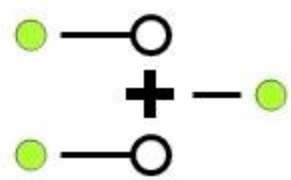
-) Consists of *Seize*, *Delay*, *Release*

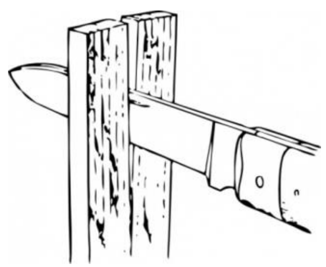
-) Capacity

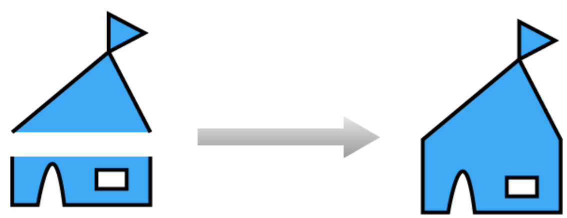
-) Timeout and preempted abort

Anylogic GUI: Blocks for DES Models









Split and **Combine** Initialize events *„Copy“ and „Join“*

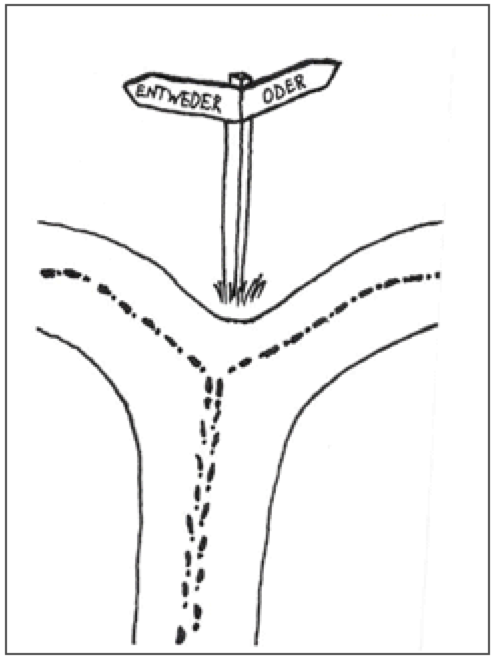
Parameters:

-) Number of copies

-) Different classes of copies possible

-) Does not forward the CLOCK

Anylogic GUI: Blocks for DES Models



SelectOutput

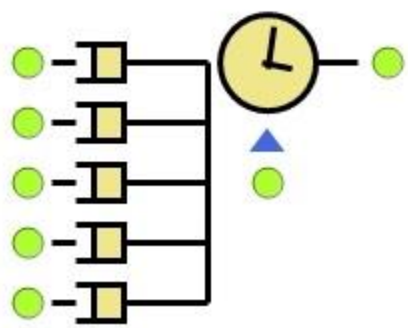
Initializes event „*Decide*"

Parameters:

-) On condition

-) On probability

Anylogic GUI: Blocks for DES Models



Assembler

Initializes event *„construction“*

Parameters:

-) Capacity of inputs

-) Delay

-) Can use resources

-) Different classes possible

Anylogic GUI: Blocks for DES Models

Conveyor

Initializes event *„conveyor“*

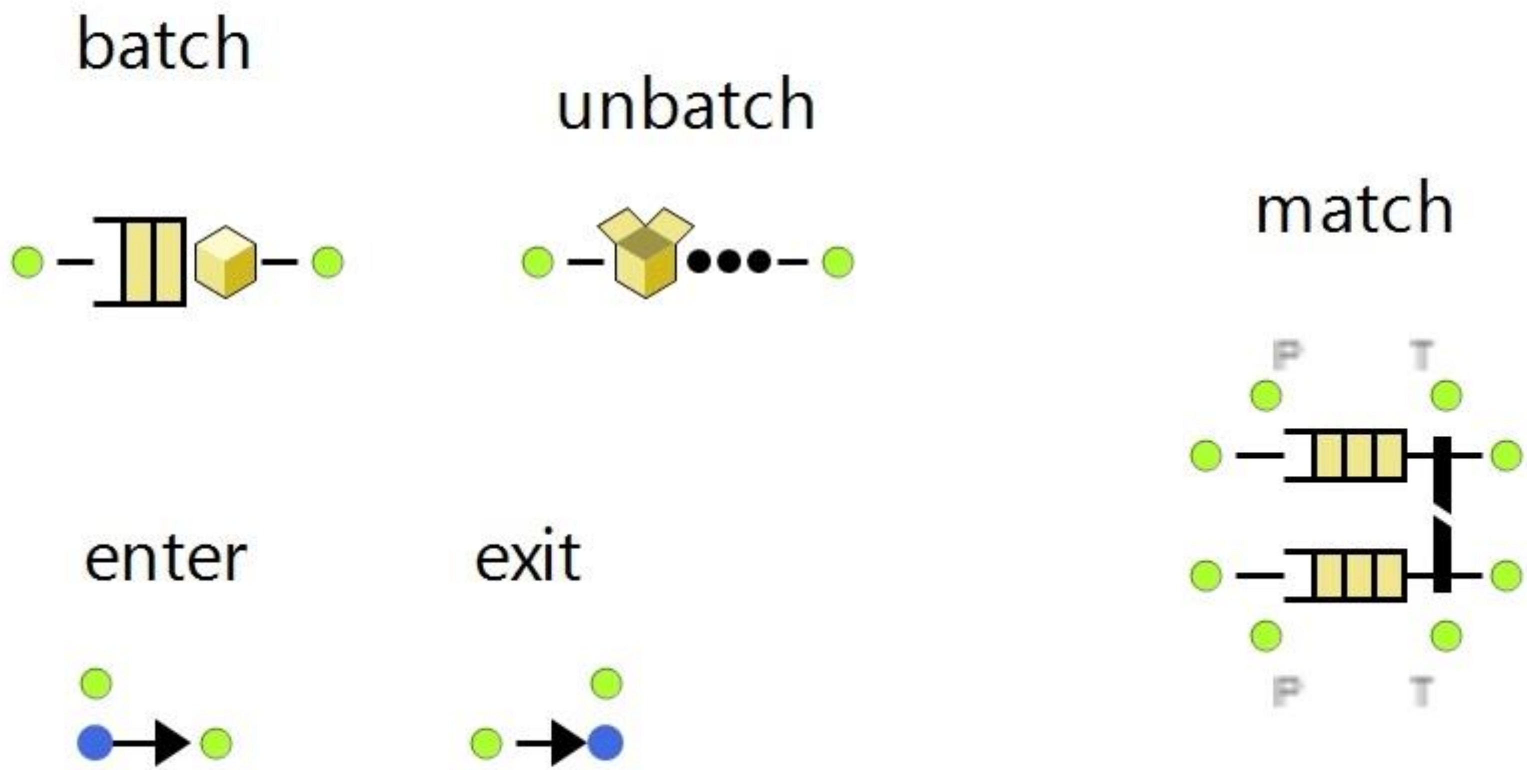
Parameters:

-) Length

-) Space between entities

-) Speed

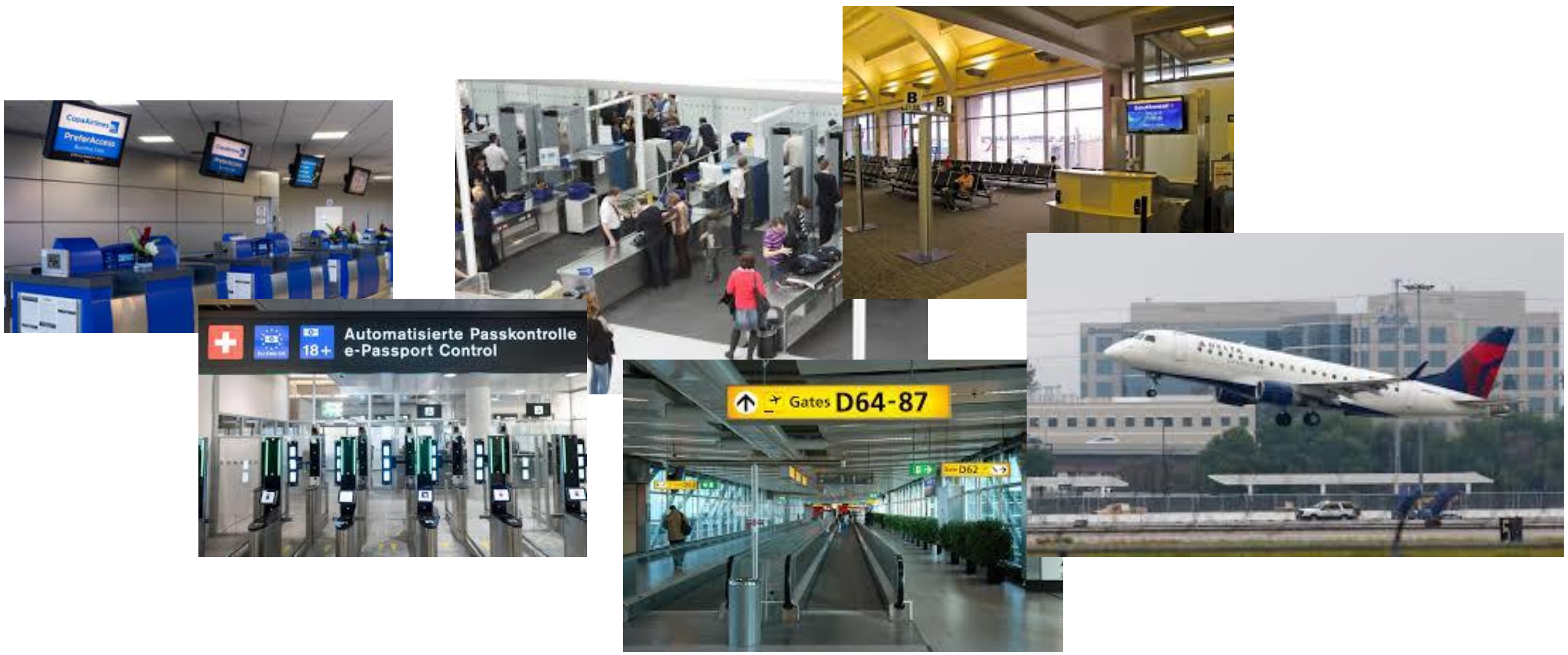
Anylogic GUI: Blocks for DES Models



Example: Airport

Research Question:

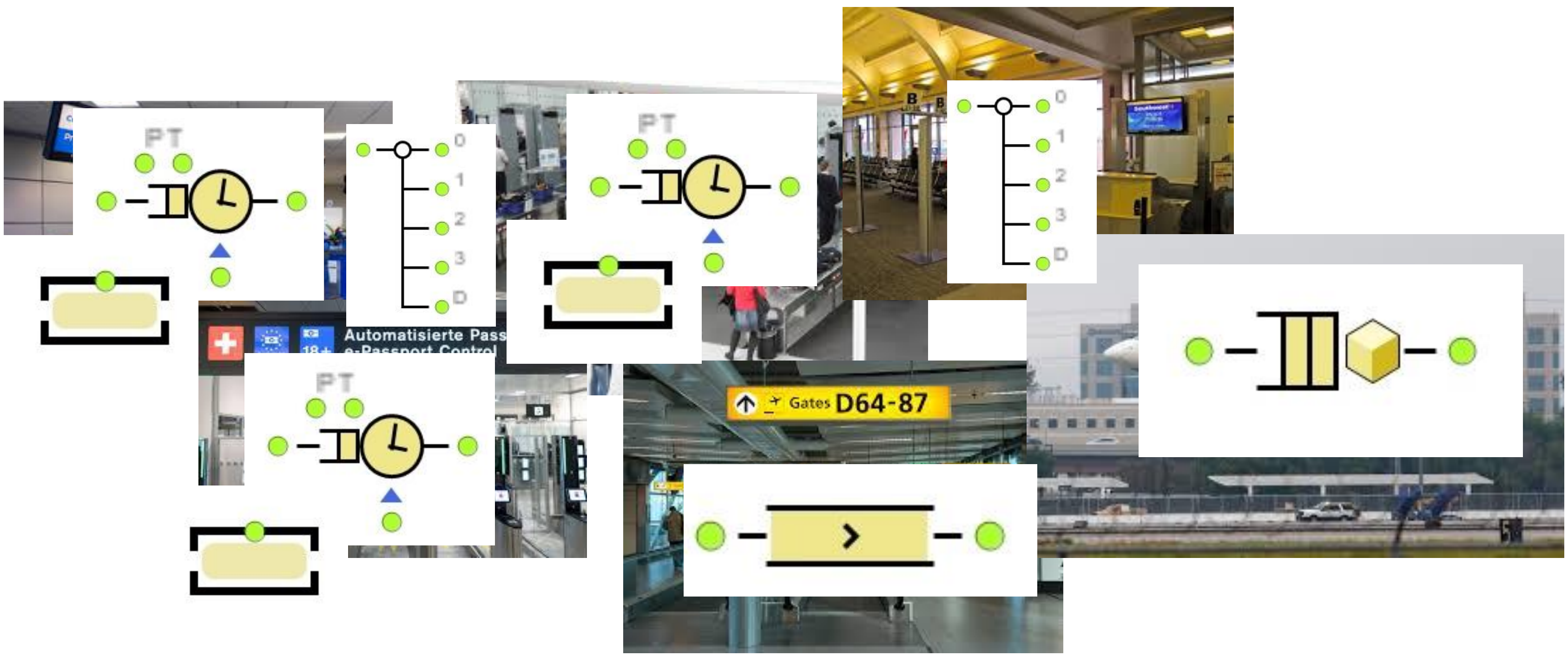
How many check-in counters, security control and counters for passport control do we need on an airport with given flight schedule?



Example: Airport

Research Question:

How many check-in counters, security control and counters for passport control do we need on an airport with given flight schedule?

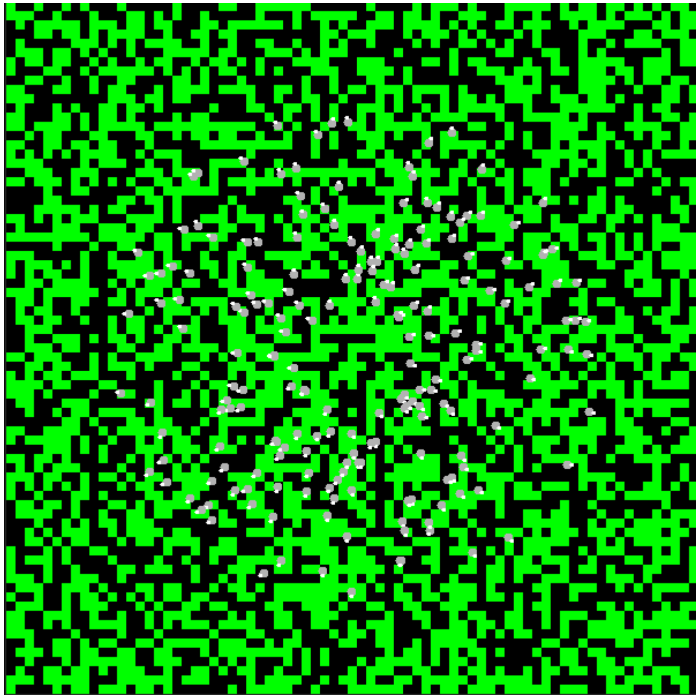


Introduction to Cellular Automata

BASIC CONCEPTS

Cellular Automata

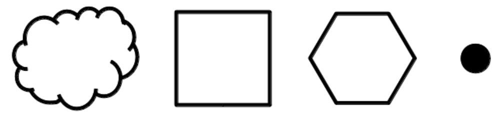
- Modelling using „cellular automata“, short CA, is a microscopic simulation method
- Cellular automata can be imagined as a coloured grid observed dynamically



Although this is a very simplified image of a CA, keep it in mind to understand the formal details of this concept

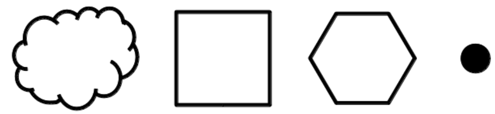
Components of a CA

- Cells



Components of a CA

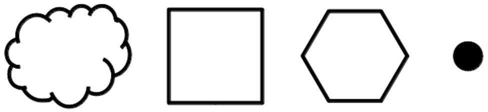
- Cells



- Notations: cell, entity, node
- Cells are passive: no internal dynamic, only container for some information
- Each cell has some state.

Components of a CA

- Cells

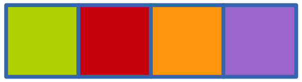


- States

A | 3.14 | (3,6,1) | 1

- State-space

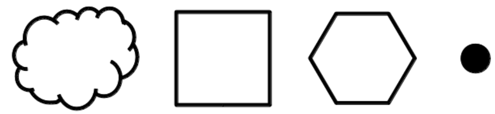
$\mathbb{R}$ $\mathbb{Z}^4$ $\{1, 2, 3\}$



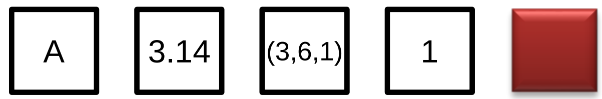
- Every Cell has a state
- There is always some space $\mathbb{S}$ that contains all possible states. It is usually called state-space.

Components of a CA

- Cells

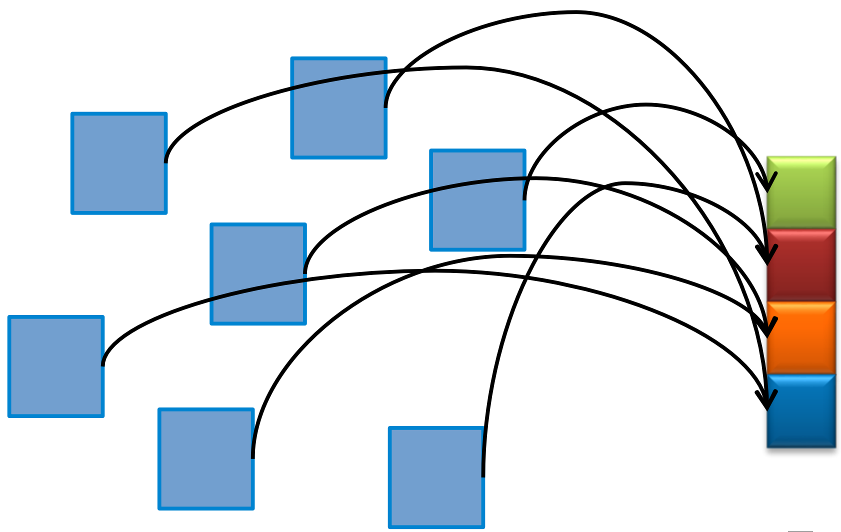


- States



- State-space

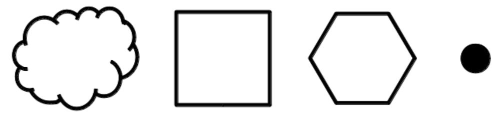
$\mathbb{R}$ $\quad \mathbb{Z}^4$ $\quad \{1,2,3\}$



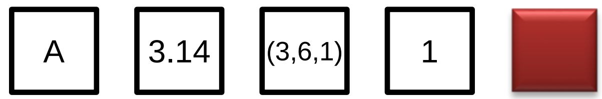
Every cell has a state from a common state-space

Components of a CA

- Cells

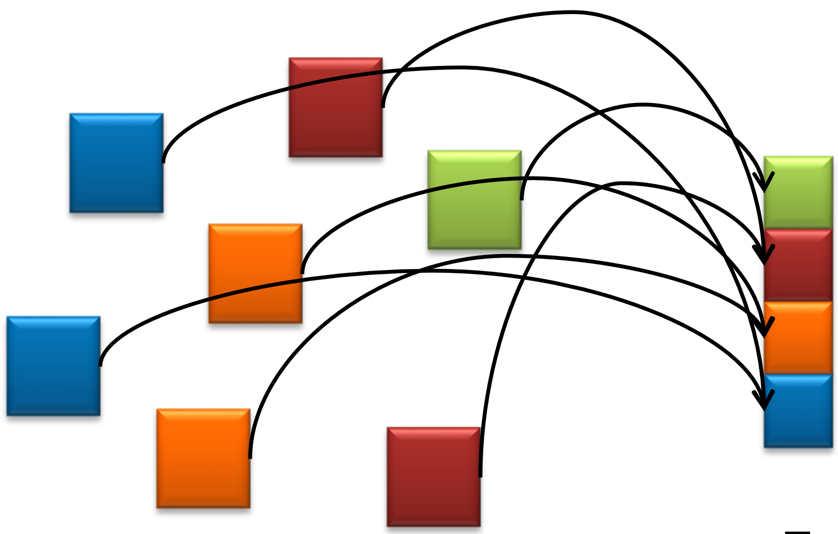


- States



- State-space

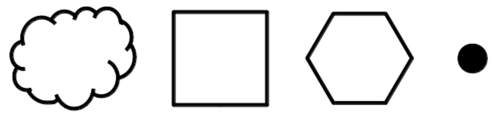
$\mathbb{R}$ $\mathbb{Z}^4$ $\{1,2,3\}$



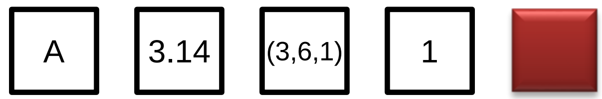
Every cell has a state from a common state-space

Components of a CA

- Cells



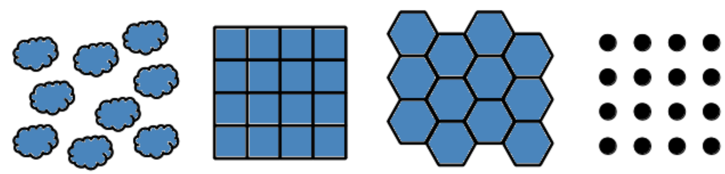
- States



- State-space

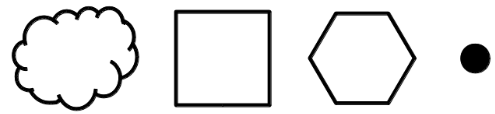
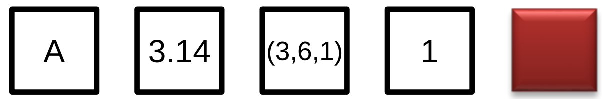
$\mathbb{R}$ $\mathbb{Z}^4$ $\{1,2,3\}$

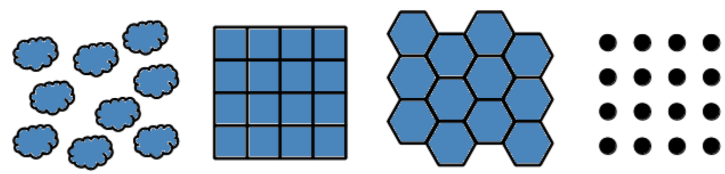
- Arrangement (Cell-space)

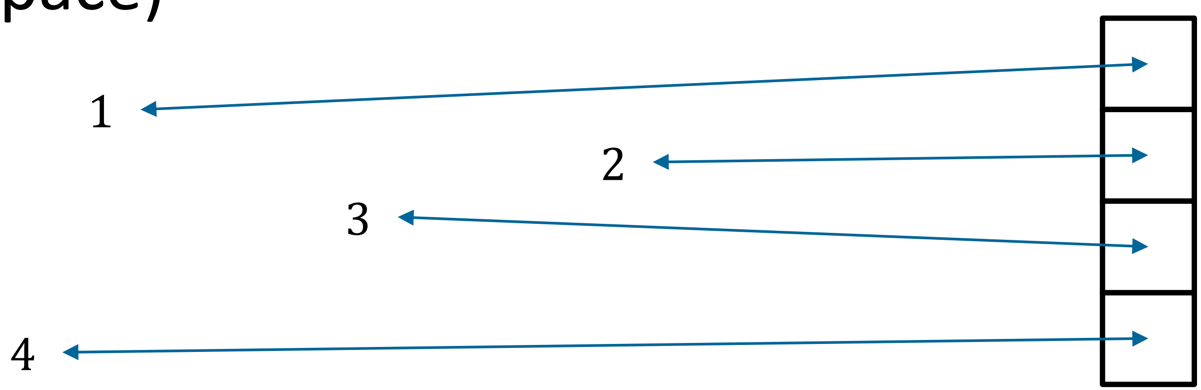


- All cells are arranged on some lattice structure: the „cell-space“ – in the simplest case, a rectangular grid.
- There is some index mapping that maps some subset of $\mathrm{I} \subset \mathbb{Z}^d$ onto each cell

Components of a CA

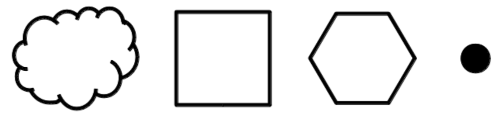
- Cells 
- States 

- State-space $\mathbb{R}$ $\mathbb{Z}^4$ $\{1,2,3\}$ 
- Arrangement (Cell-space) 

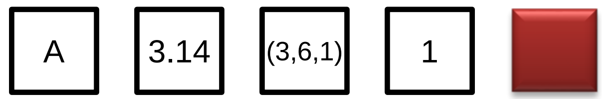


Components of a CA

- Cells



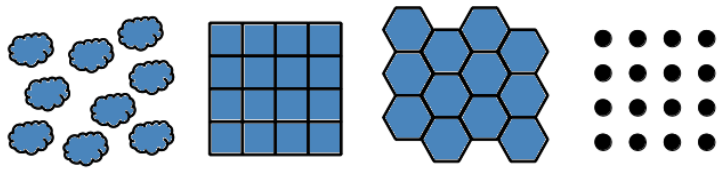
- States

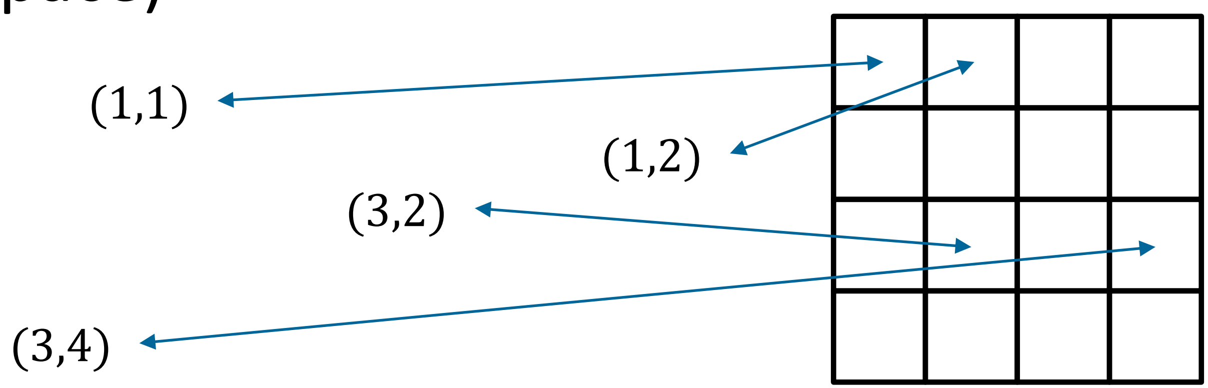


- State-space

$\mathbb{R}$ $\mathbb{Z}^4$ $\{1, 2, 3\}$

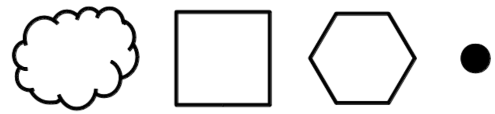
- Arrangement (Cell-space)





Components of a CA

- Cells



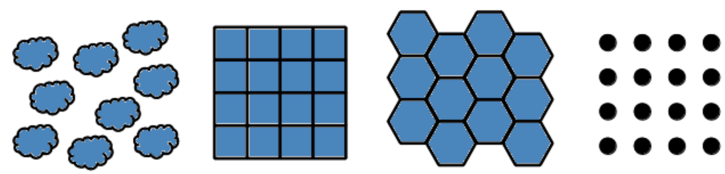
- States

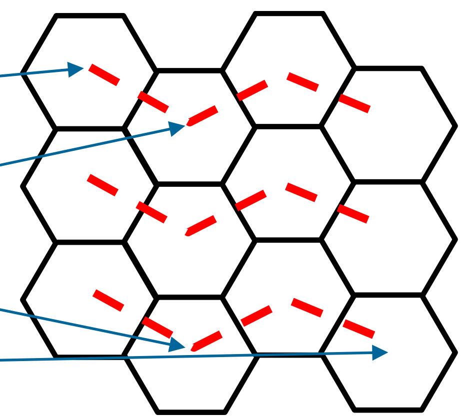
A | 3.14 | (3,6,1) | 1

- State-space

$\mathbb{R}$ $\mathbb{Z}^4$ $\{1, 2, 3\}$

- Arrangement (Cell-space)





(1,1)

(1,2)

(3,2)

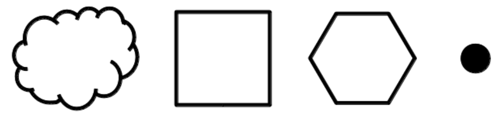
(3,4)

Sometimes indexing is not so trivial…

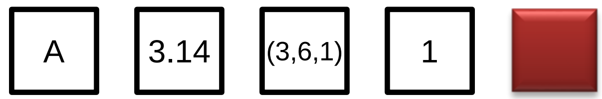
Components of a CA

TU WIEN
Mathematical Modelling and Simulation

- Cells



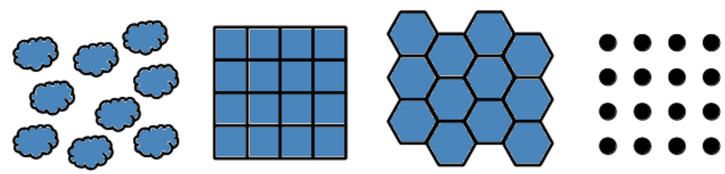
- States



- State-space

$\mathbb{R}$ $\mathbb{Z}^4$ $\{1, 2, 3\}$

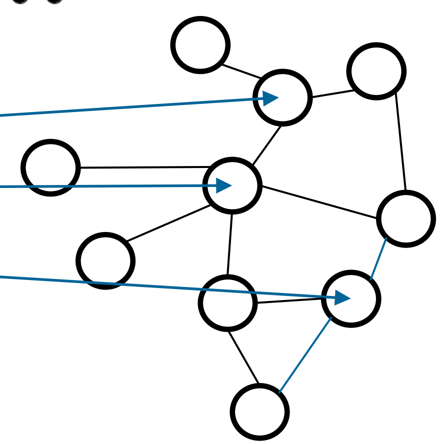
- Arrangement (Cell-space)



(1,1)

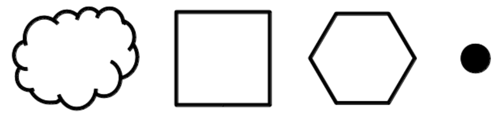
(1,2)

(3,2)



It often is, but does not necessarily have to be a natural attribute of the cell-space…

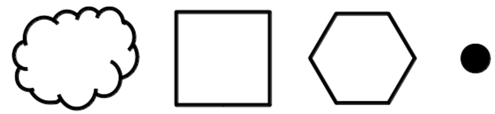
Components of a CA

- Cells 
- States A | 3.14 | (3,6,1) | 1 
- State-space $\mathbb{R}$ $\mathbb{Z}^4$ $\{1,2,3\}$ 

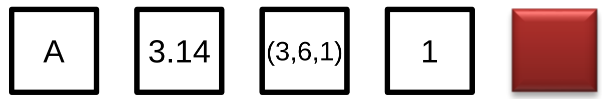
- Possible characteristics of the index set:
  - regular
  - finite or infinite
  - connected
  - multi-dimensional
- Interpretation of the index set: discretisation of a space or spatial arrangement of entities

Components of a CA

- Cells



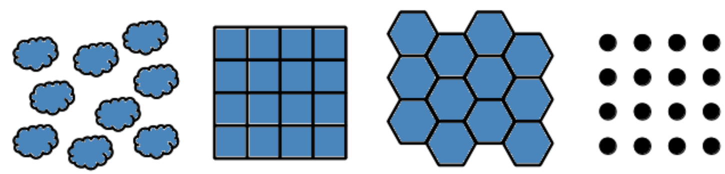
- States



- State-space

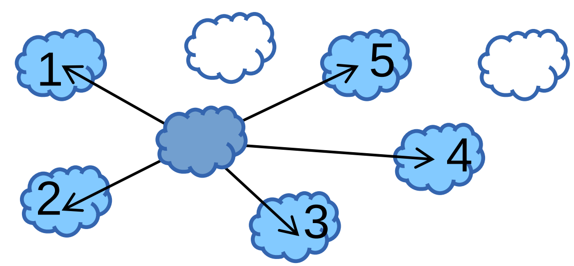
$\mathbb{R}$ $\mathbb{Z}^4$ $\{1, 2, 3\}$

- Arrangement (Cell-space)



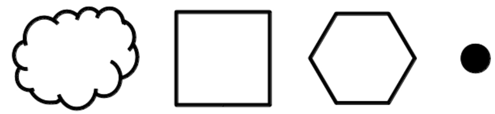
- Neighbourhood

The neighborhood of a cell *z* is an ordered set of *n* other cells $(z_1, \dots, z_n)$.

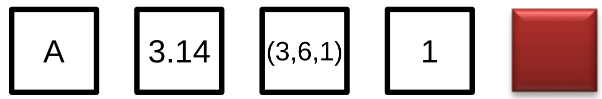


Components of a CA

- Cells



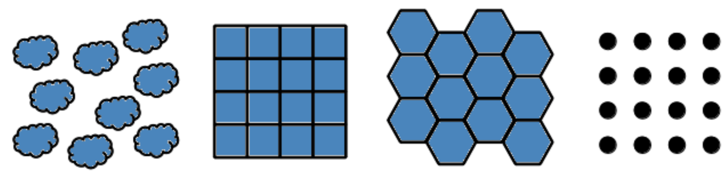
- States



- State-space

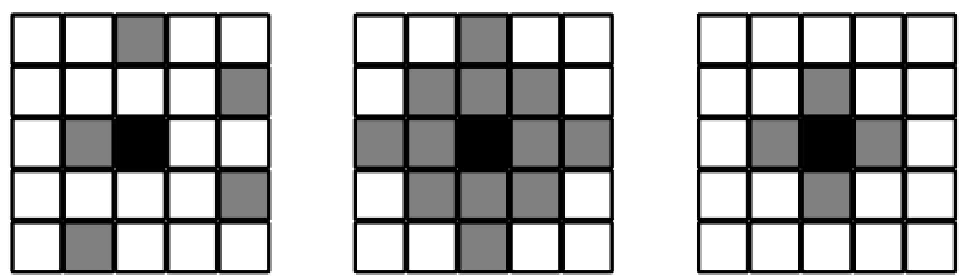
$\mathbb{R}$ $\quad \mathbb{Z}^4 \quad$ $\{1, 2, 3\}$

- Arrangement (Cell-space)



- Neighbourhood

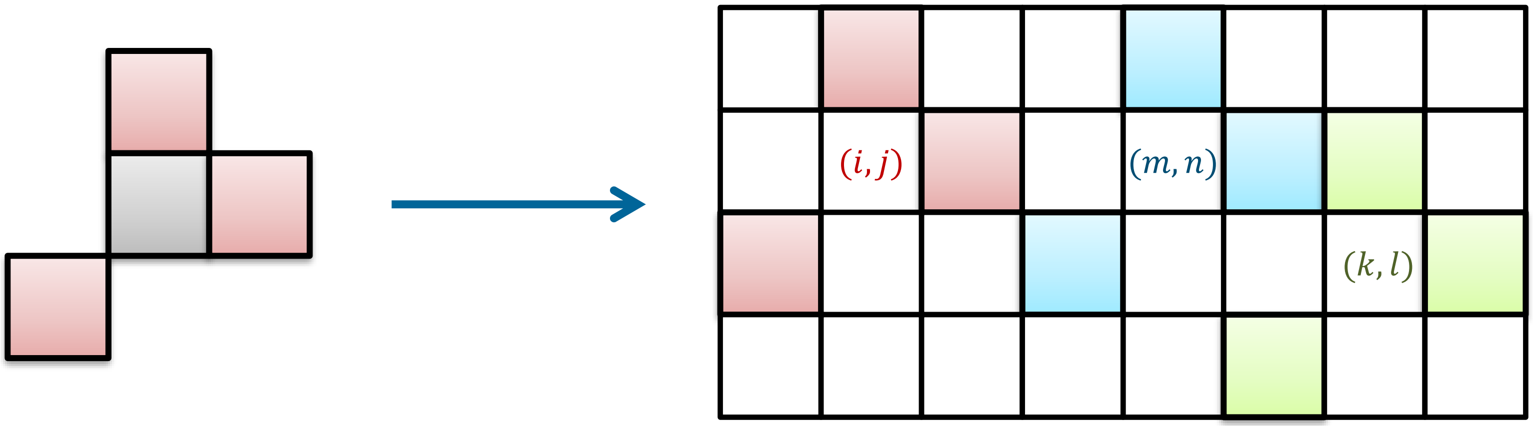
Some examples:



Neighbourhood

Mathematical Modelling and Simulation

- The neighbourhood mapping is relative to the cell's position (= index)



- Calculation of neighbouring cells by stencil: Index translations yield the positions (index) of *n* neighboring cells: $\vec{i} \mapsto (\vec{i} + \vec{t_1}, \ldots \vec{i} + \vec{t_n})$

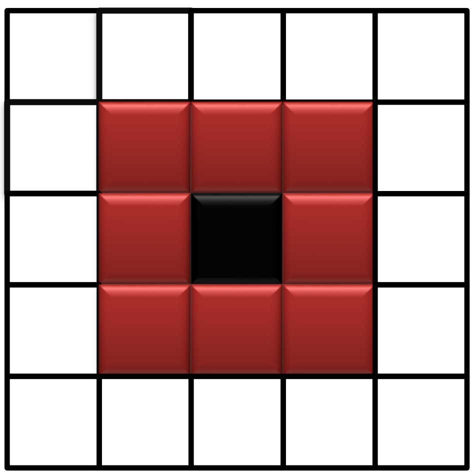
Neighbourhood

- Possible characteristics of neighbourhoods:
  - **local:** the neighbourhood consists of cells of neighboring points on the grid
  - **symmetric:** the neighborhood of cell A contains cell B if and only if the neighborhood of cell B contains cell A

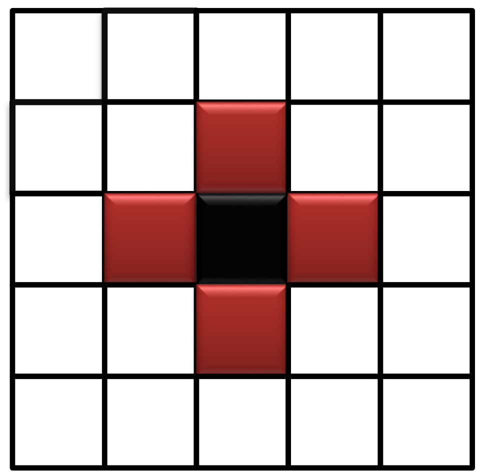
Neighbourhood

- Classic, popular neighborhoods

Moore neighborhood

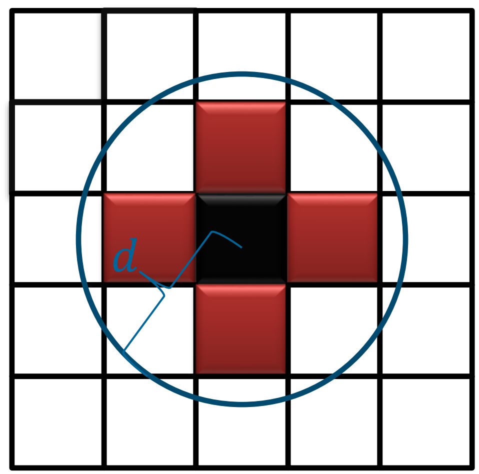


Von-Neumann neighborhood



Neighbourhood by distance:

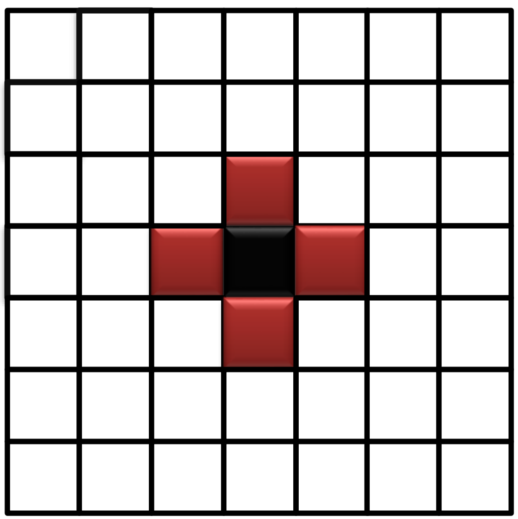
$$\vec{i} \to \{\vec{j} : |\vec{i} - \vec{j}| < d\}$$



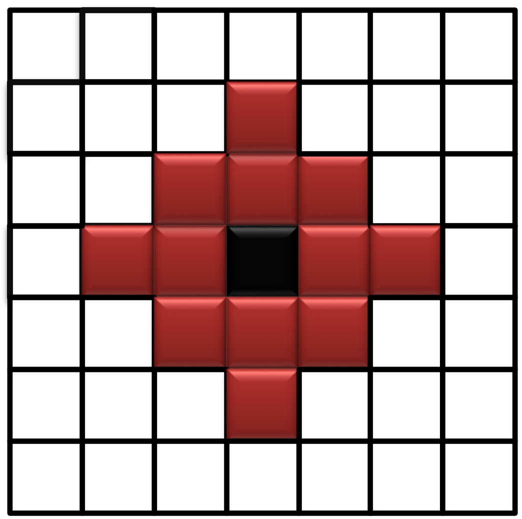
Neighbourhood

- Von Neumann/Moore Neighbourhood of higher order

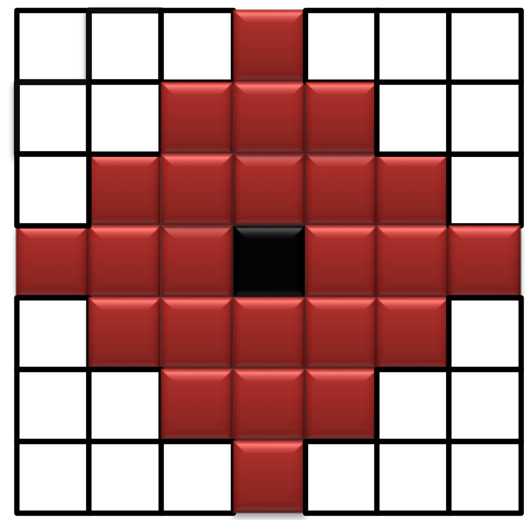
Von-Neumann neighborhood 1st order



Von-Neumann neighborhood 2nd order

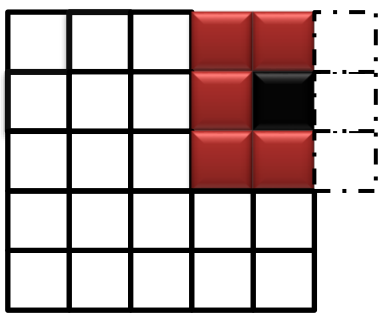


Von-Neumann neighborhood 3rd order

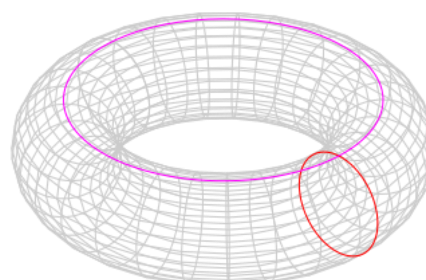


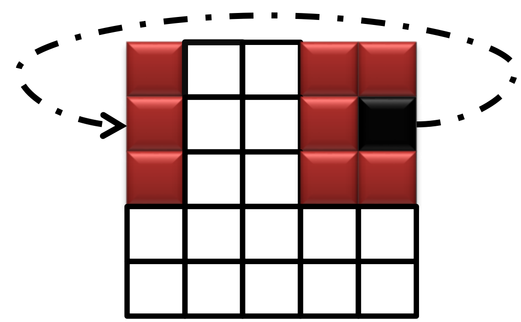
Neighbourhood

- The index set is limited → either incomplete neighborhoods for cells near the borders $(z_1, z_2, \emptyset, z_4 \ldots, z_n)$….



…or other compensation idea

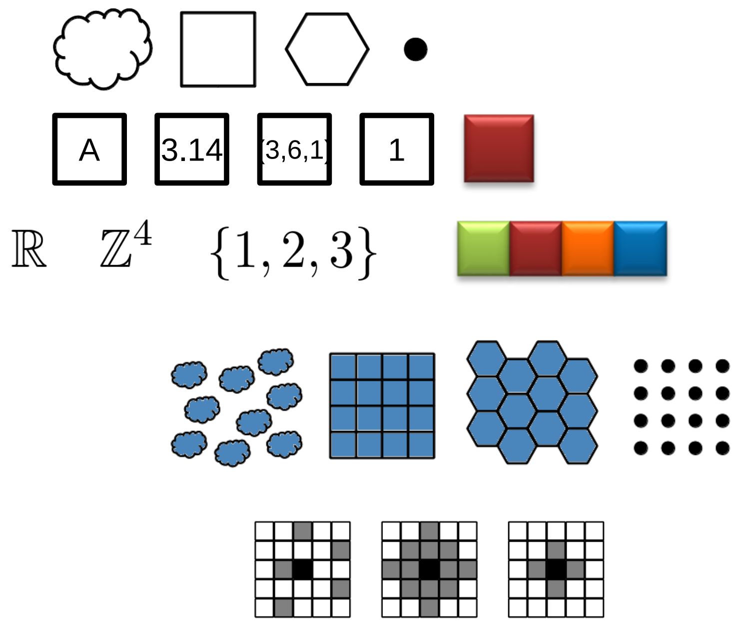




Periodic Boundary Conditions (Torus)

Components of a CA

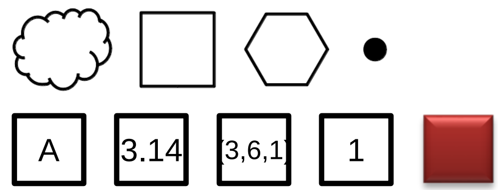
- Cells
- States
- State-space
- Arrangement (Cell-space)
- Neighbourhood
- Update Rule



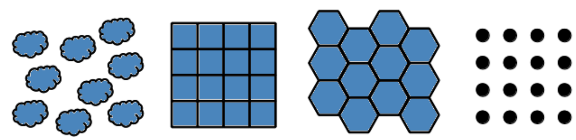
Some rule, that simultaneously updates all states of all cells of the CA.
Maps all states of a cell’s neighbourhood to a new state for the cell.

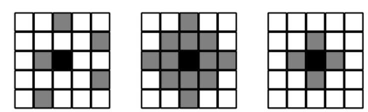
Components of a CA

- Cells
- States
- State-space
- Arrangement (Cell-space)
- Neighbourhood
- Update Rule



$\mathbb{R}$ $\quad \mathbb{Z}^4$ $\quad \{1,2,3\}$





$$f(s, s_1, \dots, s_n) = s_{new}$$

state of the cell

state of the (ordered) neighbors

new state of the cell

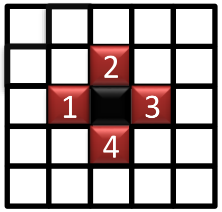
Stochastic CAs have stochastic updates!

Update Rule

- Example:

Neighbourhood = Von Neumann

$$f(s, s_1, s_2, s_3, s_4) = \sum s \ (mod\ 4)$$



Old state of the CA

1	2	1	0	2	1	1
2	3	1	2	3	1	1
0	2	1	2	0	2	1
3	2	1	1	1	5	1
0	1	2	0	2	2	0
2	0	1	1	1	3	1
2	2	0	2	3	1	2

New state of the CA

Update Rule

- Example:

Neighbourhood = Von Neumann

$$f(s, s_1, s_2, s_3, s_4) = \sum s \ (mod\ 4)$$

		2		
	1		3	
		4		

Old state of the CA

1	2	1	0	2	1	1
2	3	1	2	3	1	1
0	2	1	2	0	2	1
3	2	1	1	1	5	1
0	1	2	0	2	2	0
2	0	1	1	1	3	1
2	2	0	2	3	1	2

$$f(s, s_1, s_2, s_3, s_4) =$$
$$= 1 + 1 + 2 + 1 + 0\ (mod 4) =$$
$$= 5\ (mod\ 4) = 1$$

New state of the CA

			1			

Update Rule

- Example:

Neighbourhood = Von Neumann

		2		
	1		3	
		4		

$$f(s, s_1, s_2, s_3, s_4) = \sum s \ (mod\ 4)$$

Old state of the CA

1	2	1	0	2	1	1
2	3	1	2	3	1	1
0	2	1	2	0	2	1
3	2	1	1	1	5	1
0	1	2	0	2	2	0
2	0	1	1	1	3	1
2	2	0	2	3	1	2

$$f(s, s_1, s_2, s_3, s_4) =$$
$$= 1 + 1 + 2 + 3 + 3\ (mod 4) =$$
$$= 10\ (mod\ 4) = 2$$

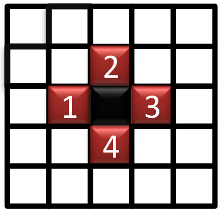
New state of the CA

			1			
					2	

Update Rule

- Example:

Neighbourhood = Von Neumann



$$f(s, s_1, s_2, s_3, s_4) = \sum s \ (mod\ 4)$$

Old state of the CA

1	2	1	0	2	1	1
2	3	1	2	3	1	1
0	2	1	2	0	2	1
3	2	1	1	1	5	1
0	1	2	0	2	2	[illegible]
2	0	1	1	1	3	[illegible]
2	2	0	2	3	1	[illegible]

$$f(s, s_1, s_2, \emptyset, s_4) =$$
$$= 1 + 1 + 1 + 1\ (mod 4) =$$
$$= 4\ (mod\ 4) = 0$$

New state of the CA

						0
			1			
					2	

The update function needs to be capable to deal with incomplete neighbourhoods as well

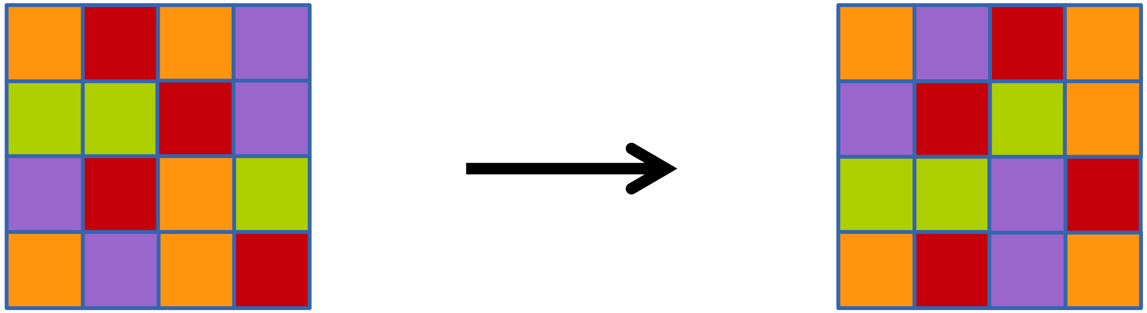
Updates

- Updates happen for all cells simultaneously.

  Why?

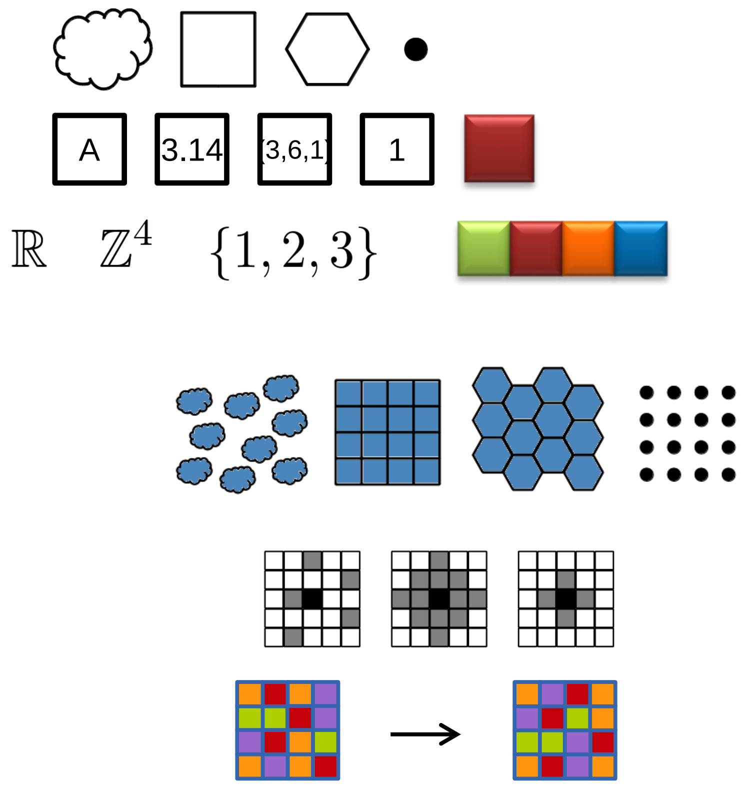
  – Neighborhoods are all computed from the same system state

  – Update order of cells is irrelevant



Components of a CA

- Cells
- States
- State-space
- Arrangement (Cell-space)
- Neighbourhood
- Update Rule
- Iterations



Iteratively apply the update rule on the complete CA finally leads to a simulation model

Iterations

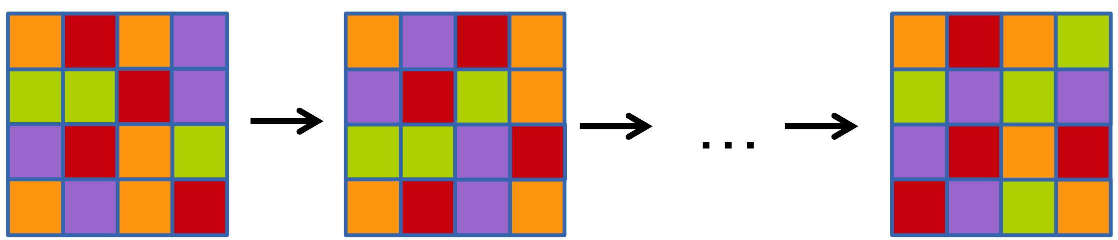
Mathematical Modelling and Simulation

- Define discrete, equidistant time points (all time steps between time points are of the same length): $t_0, t_1, \ldots, t_n$
- Every update of states brings the model to the next time point

➢ Cellular Automaton (CA)

Iterations

- Tasks for one iterations
  - Compute the neighbors of all cells
  - Determine states of all cells, and states of all neighbours of all cells
  - Compute state updates for all cells and store them
  - Apply the updated states for all cells

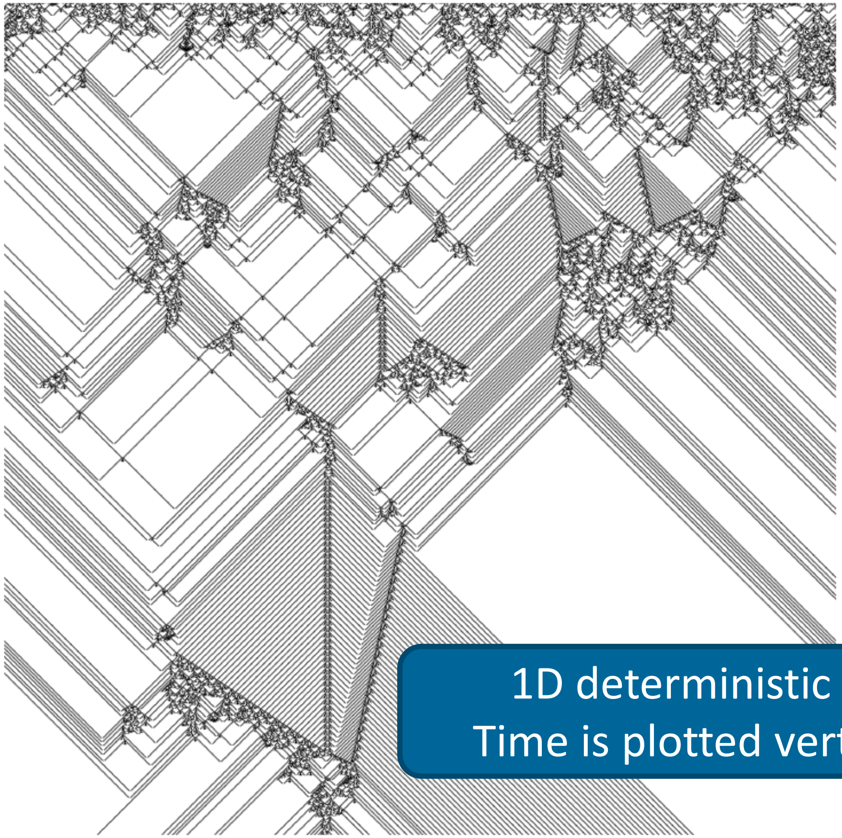


time t_0 time t_1 … time t_n

Properties of CA models

Cellular Automata are microscopic simulation models that are capable of producing almost arbitrarily complex, up to chaotic, behaviour.

They are, hence, not only a very powerful, but also a very dangerous modelling approach with respect to validity.



1D deterministic CA!
Time is plotted vertically

Properties of CA models

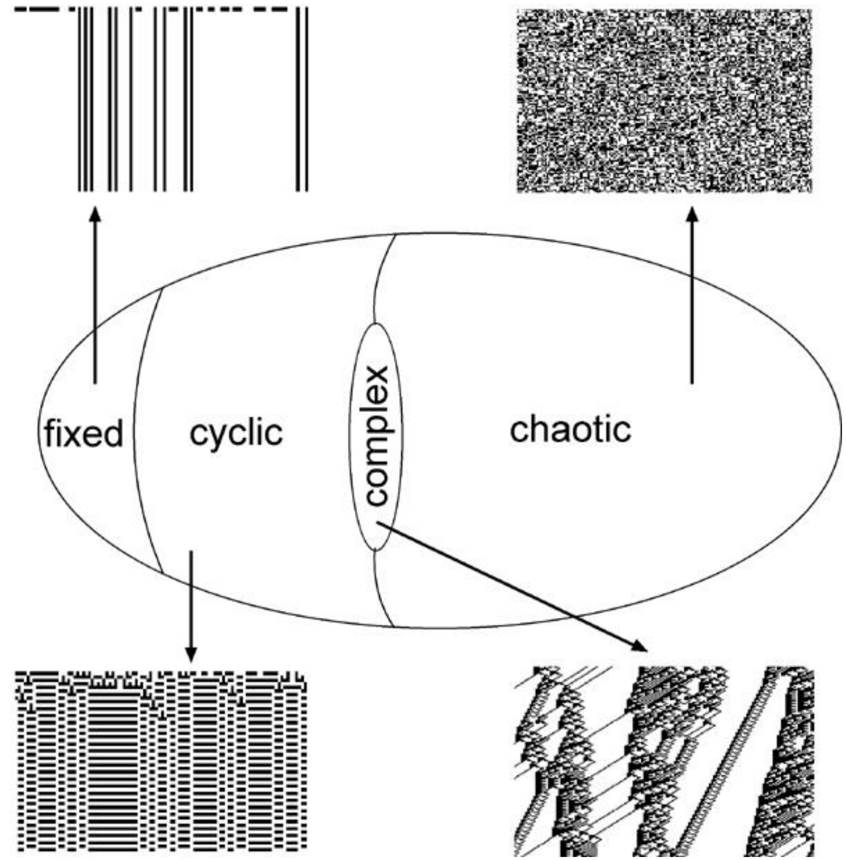
Mathematical Modelling and Simulation

Cellular Automata are microscopic simulation models that are capable of producing almost arbitrarily complex, up to chaotic, behaviour.

Stephen Wolfram
(A New Kind of Science, 2002)
stated that CAs may have one of the four types of behavour:

fixed, cyclic, complex, chaotic

Chris Langton developed the schematic to the right.



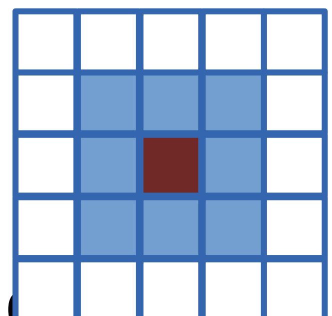
Mathematical Modelling and Simulation

Example:

CONWAY’S GAME OF LIFE

Conway's Game of Life

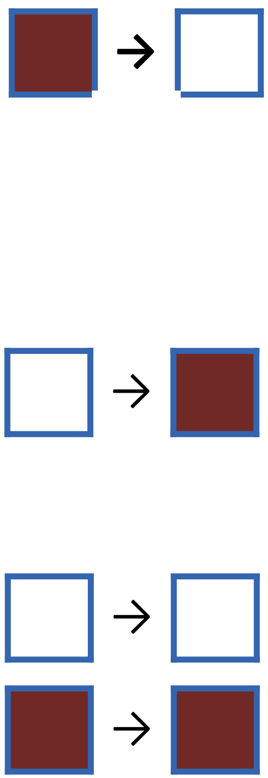
- Cells on a 2-dimensional, rectangular or infinite lattice: $I = (1,2,\dots a) \times (1,2,\dots,b)$ or on $I = \mathbb{Z}^2$.
- Set of states: $\mathbb{S} = (alive \quad , dead \quad)$
- Moore neighborhood



Index translations:

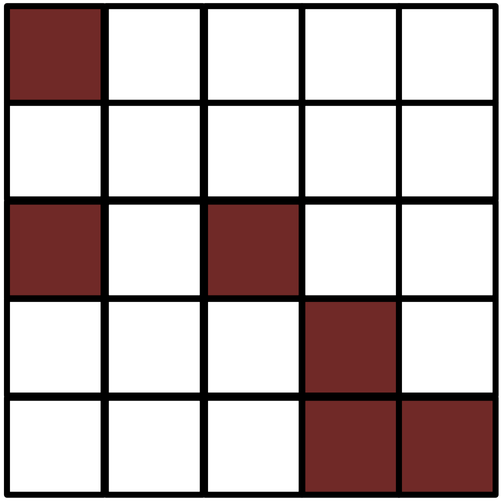
$$\left(\binom{1}{0},\binom{1}{1},\binom{0}{1},\binom{-1}{1},\binom{-1}{0},\binom{-1}{-1},\binom{0}{-1},\binom{1}{-1}\right)$$

Conway‘s Game of Life

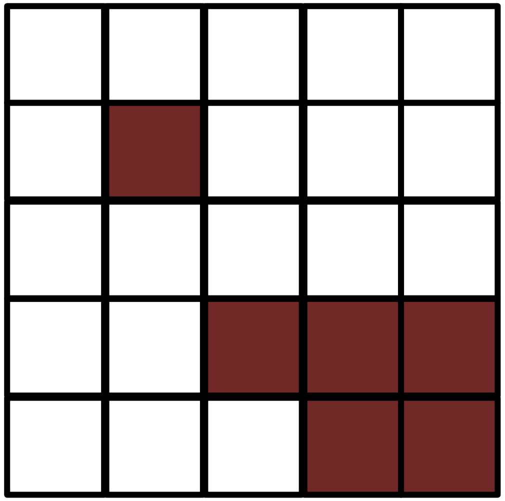


- Update rules:
  - An alive cell with fewer than two or more than three alive neighbors dies (“under-population” or “overcrowding”)
  - A dead cell with exactly three alive neighbors becomes alive (“reproduction”)
  - Cells keep their state in any other case

Conway's Game of Life



time t=0



time t=1

Conway's Game of Life

- Designed by John Horton Conway, 1970
- Why "Game of Life"?
  - Teaching purposes
  - Academic competitions
  - Fundamental/methodological research
  - Game → figures

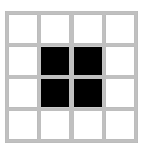
Probably ***worst*** example for a Cellular Automata simulation model,…

…but probably the ***best*** example to show the concepts of CAs.

Conway‘s Game of Life

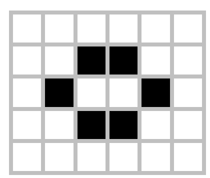
Pattern analysis of the Game of Life became its own science (although its applicability can be doubted).

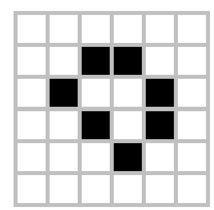
Static figures



Block

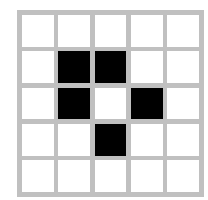
Beehive





Loaf

Boat



Source: http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life

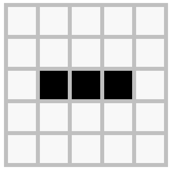
Conway's Game of Life

Pattern analysis of the Game of Life became its own science (although its applicability can be doubted).

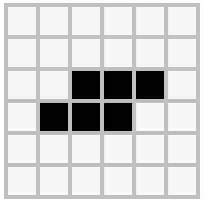
Oscillators

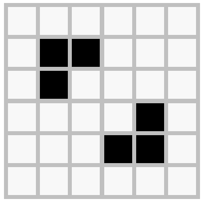
Toad (period 2)

Pulsar (period 3)

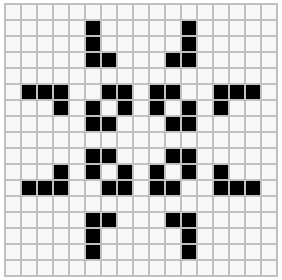


Blinker (period 2)





Beacon (period 2)

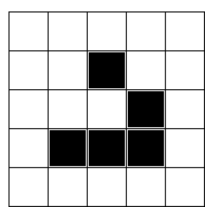


Source: http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life

Conway's Game of Life

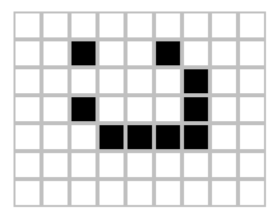
Pattern analysis of the Game of Life became its own science (although its applicability can be doubted).

Gliders (moving objects)



Glider

Lightweight spaceship (LWSS)



Source: http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life

Conway's Game of Life

TU WIEN Mathematical Modelling and Simulation

Pattern analysis of the Game of Life became its own science (although its applicability can be doubted).

As it seemed as if any starting configuration of the GoL resulted in a static or oscillating end-configuration, Conway offered a price of 50$ for a pattern that resulted in an infinitely growing population.

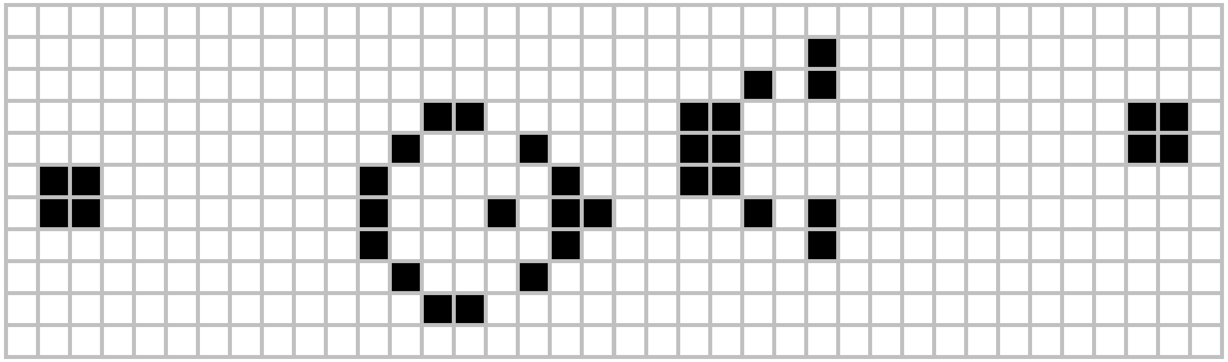
Source: http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life

Conway‘s Game of Life

Pattern analysis of the Game of Life became its own science (although its applicability can be doubted).

Bill Gosper‘s answer:

Gosper Glider Gun



Source: http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life

Example

NAGEL SCHRECKENBERG MODEL

Application Example: Traffic Simulation

Mathematical Modelling and Simulation

Nagel-Schreckenberg-Model

- discretisation of a road or motorway into cells of approximately 4m
- possible states:
  - $s = 0$: no vehicle
  - $s > 0$: speed of vehicle
- update rules (implicitly defined!):
  - accelerate: **IF** $v < v\_max$ **AND** next vehicle $v + 1$ cells away **THEN** $v(t+1) = v(t) + 1$
  - brake: **IF** next vehicle j cells away **AND** $j < v$ **THEN** $v(t+1) = j - 1$
  - randomisation: $v(t+1) = v(t) - 1$ **with a certain probability**
  - movement: $s(t+1) = s(t) + v(t)$

Application Example: Traffic Simulation

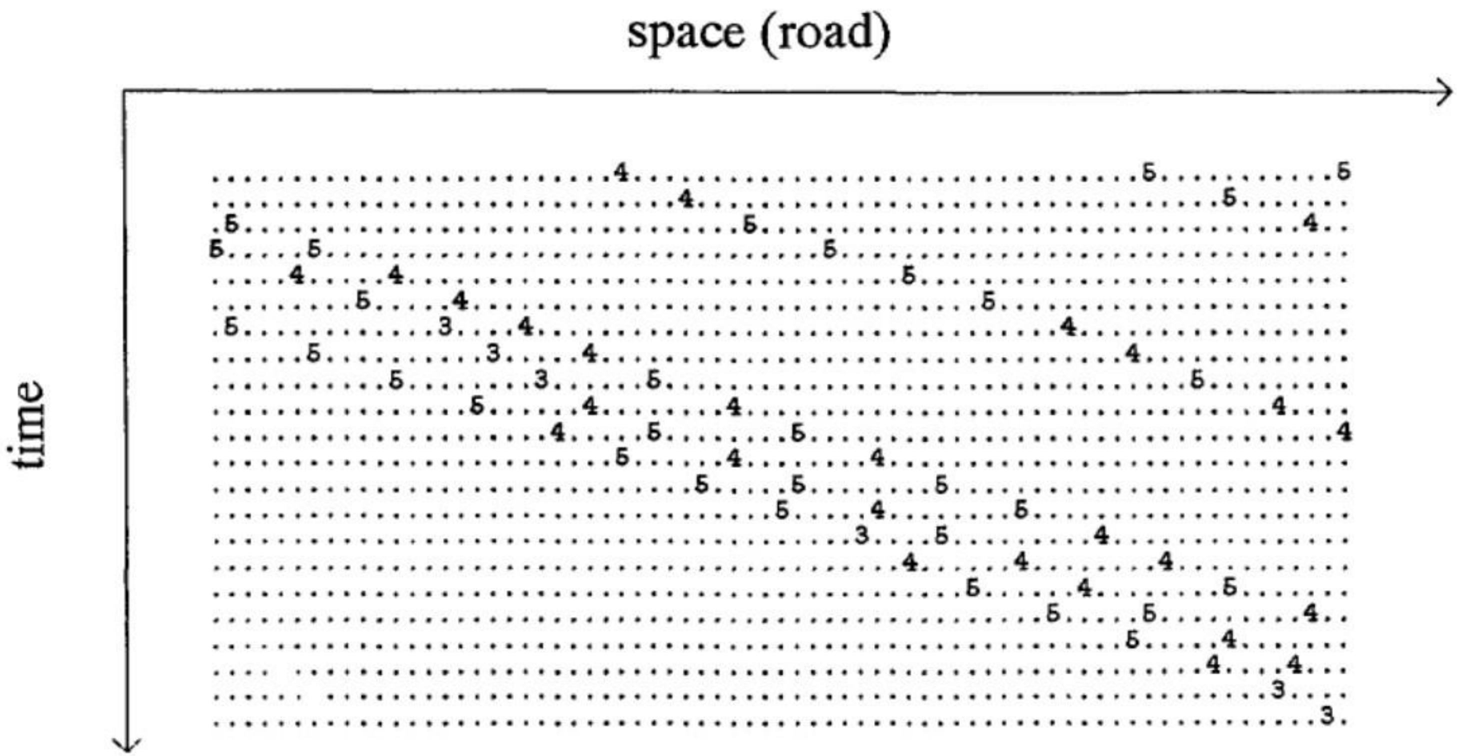


Fig.1. — Simulated traffic at a (low) density of 0.03 cars per site. Each new line shows the traffic lane after one further complete velocity-update and just before the car motion. Empty sites are represented by a dot, sites which are occupied by a car are represented by the integer number of its velocity. At low densities, we see undisturbed motion.

Application Example: Traffic Simulation

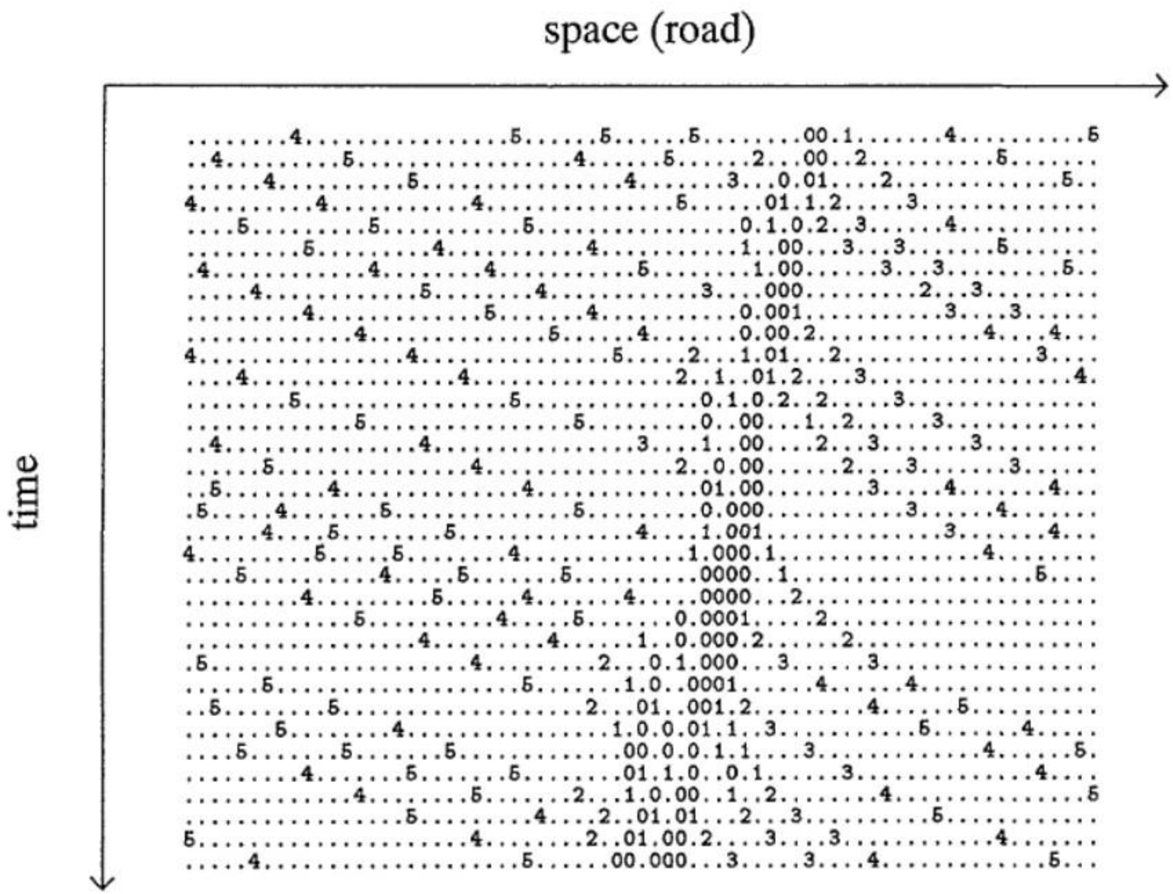


Fig.2. — Same picture as figure 1, but at a higher density of 0.1 cars per site. Note the backward motion of the traffic jam.

Application Example: Traffic Simulation

Mathematical Modelling and Simulation

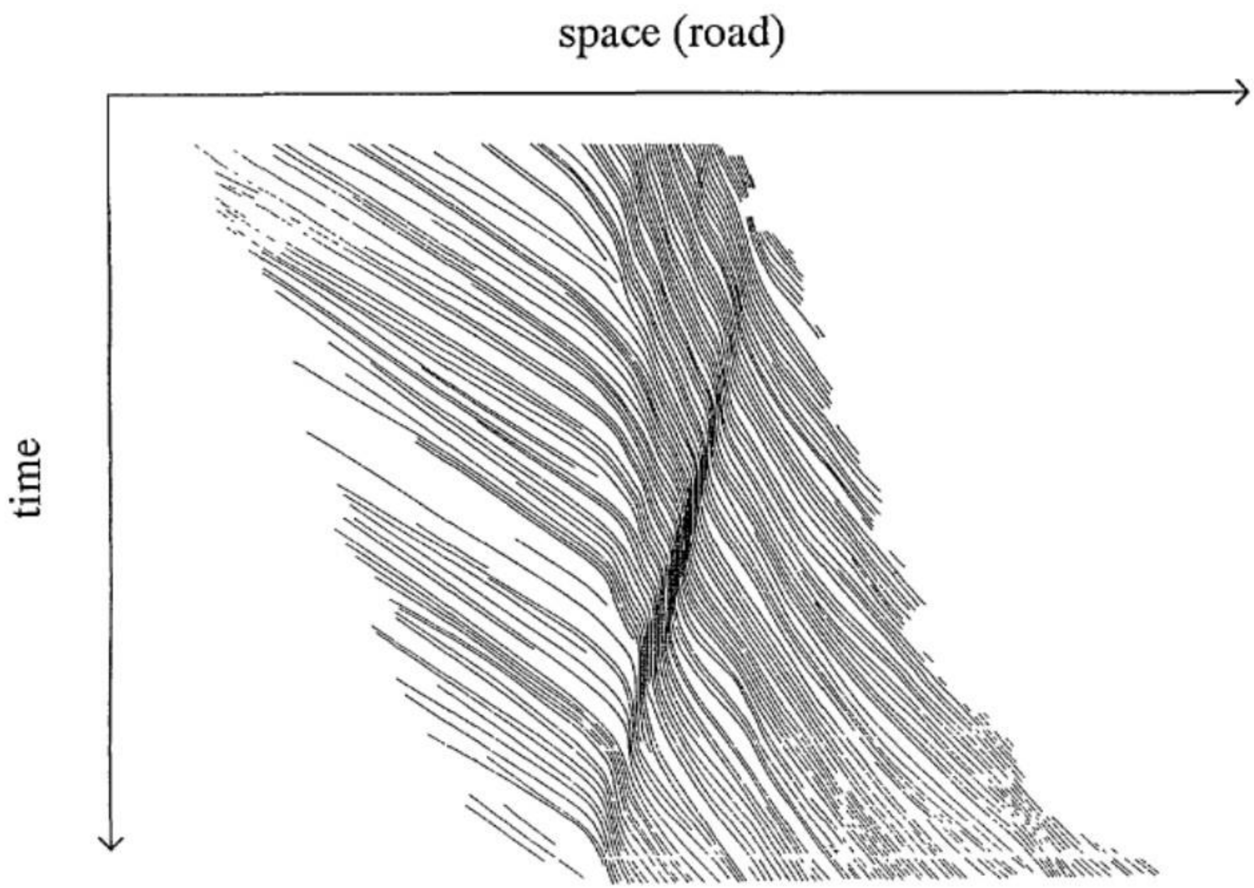


Fig.3. — Space-time-lines (trajectories) for cars from Aerial Photography (after [16]). Each line represents the movement of one vehicle in the space-time-domain.

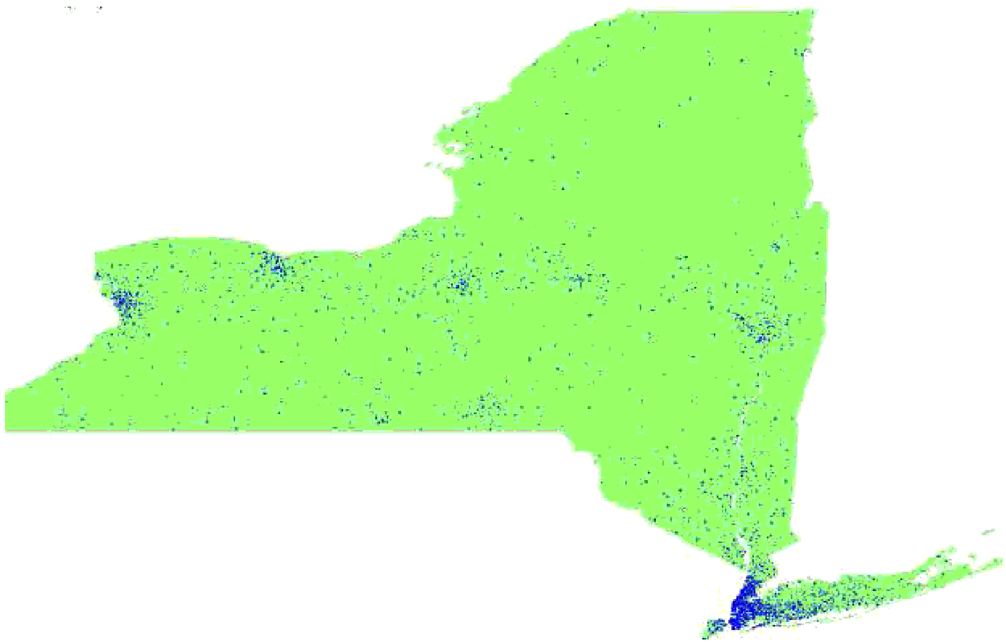
Example

DYNAMIC MAPS

Dynamic Cartography

Mathematical Modelling and Simulation

- map shows relation between sizes
- The dots symbolises cancer patients



Dynamic Cartography

Mathematical Modelling and Simulation

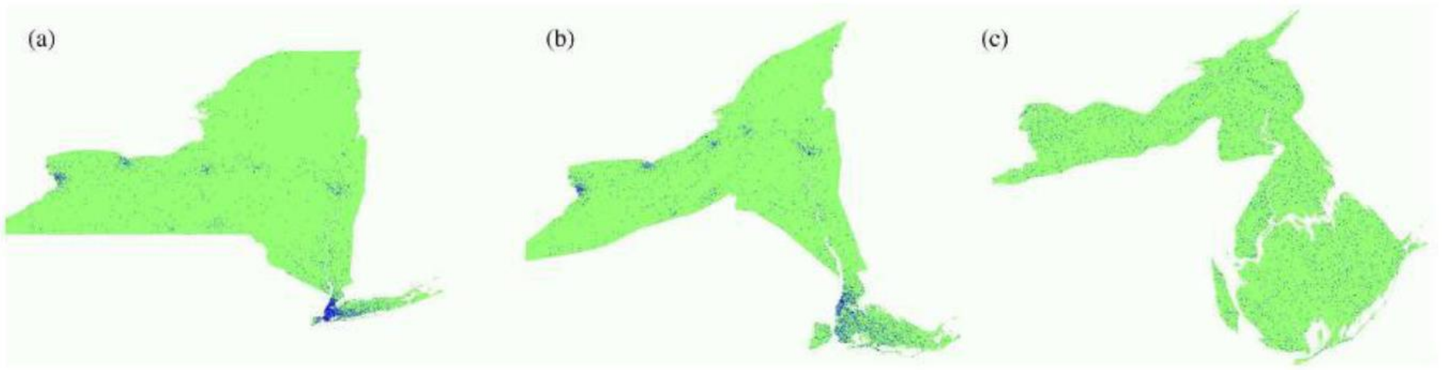


Figure 2: Visualization of lung cancer cases among males in the state of New York 1993–1997. Each dot represents ten cases, randomly placed within the zip-code area of occurrence. (a) The original map. (b) A cartogram using a coarse-grained population density with $\sigma = 0.3°$. (c) A cartogram using a much finer-grained population density ($\sigma = 0.04°$). (Data from the New York State Department of Health.)

Dynamic Cartography

Mathematical Modelling and Simulation

- Amount of cancer patients spread equally to squares in each region (e.g. staats)
- Diffusion from places with high density to low
- Diffusion continues until the density is equal distributed
- Regions with higher density grow, others shrink

Dynamic Cartography

Mathematical Modelling and Simulation

Neumann model

1,5	1,5	2	1,5
1,5	2	2	1,5
1,5	1,5	2	2
1,5	1,5	2	1,5

5	0,9	0,9	0,9
0,9	0,9	0,9	0,9
5	0,9	0,9	0,9
5	0,9	0,9	0,9

Density depending model

1,5	1,5	2	1,5
1,5	2	2	1,5
1,5	1,5	2	2
1,5	1,5	2	1,5

2,5	1,2	1,2	2,5
1,2	1,2	1,2	1,2
2,5	1,2	1,2	1,2
2,5	2,5	1,2	2,5

Moore model

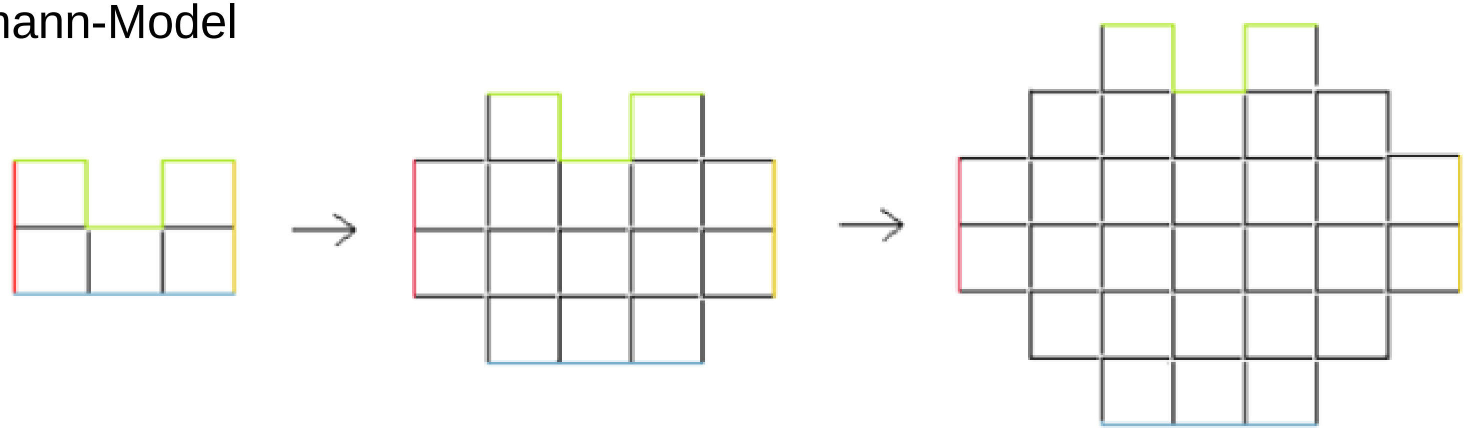
1,5	1,5	2	1,5
1,5	2	2	1,5
1,5	1,5	2	2
1,5	1,5	2	1,5

0,8	0,8	0,8	0,8
0,8	0,8	0,8	0,8
0,8	0,8	0,8	0,8
5	0,8	0,8	0,8

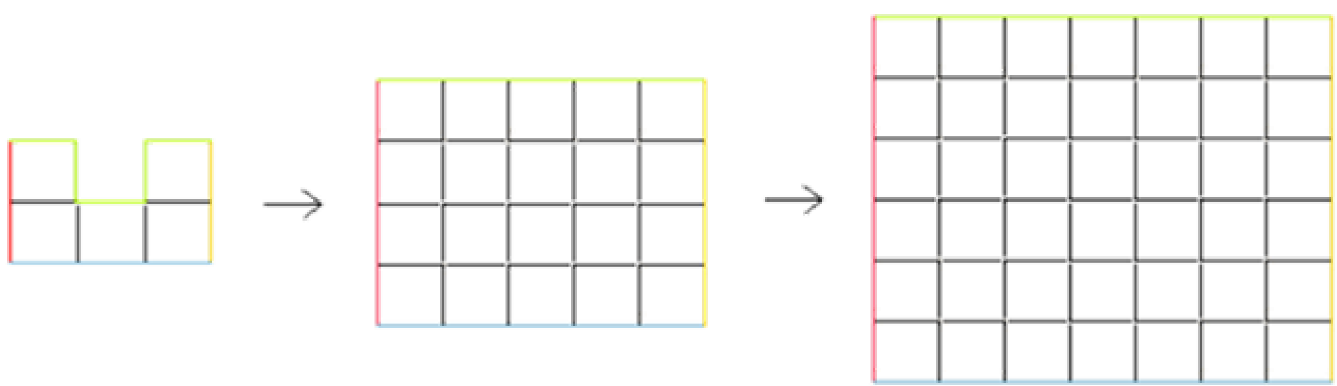
Dynamic Cartography - Boundaries

Mathematical Modelling and Simulation

Neumann-Model



Moore-Model



Dynamic Cartography - Results

Mathematical Modelling and Simulation

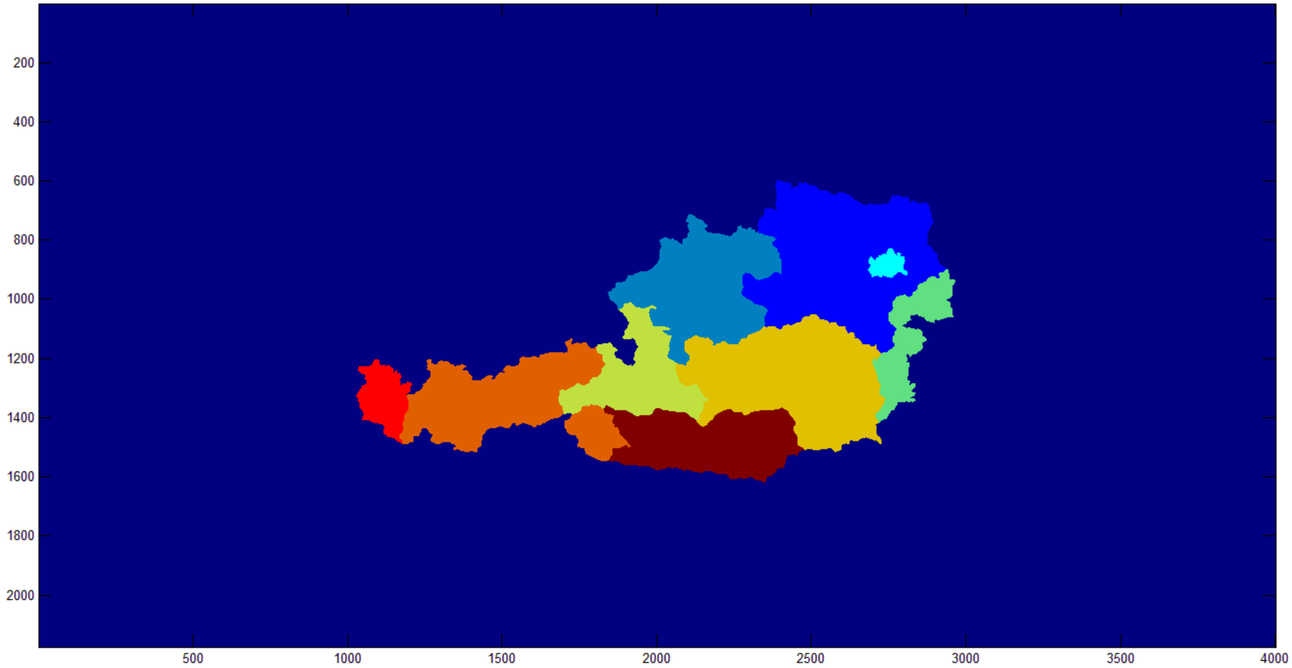
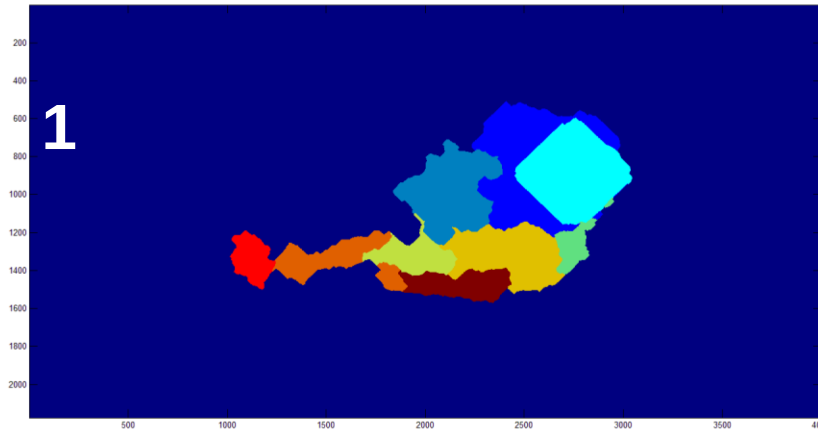
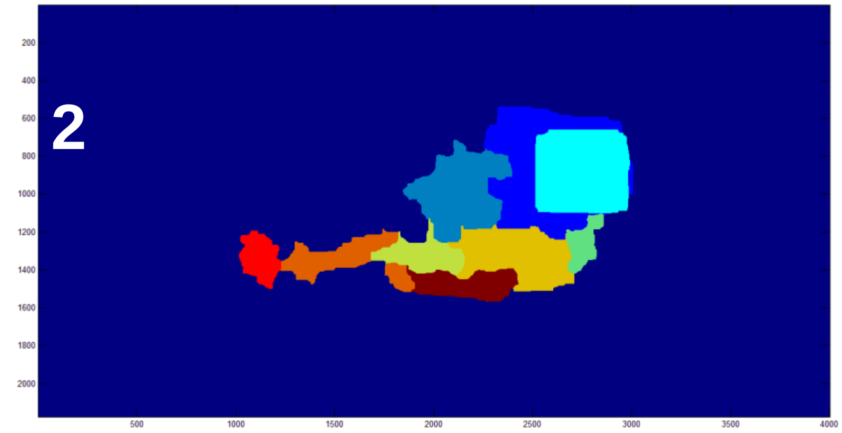


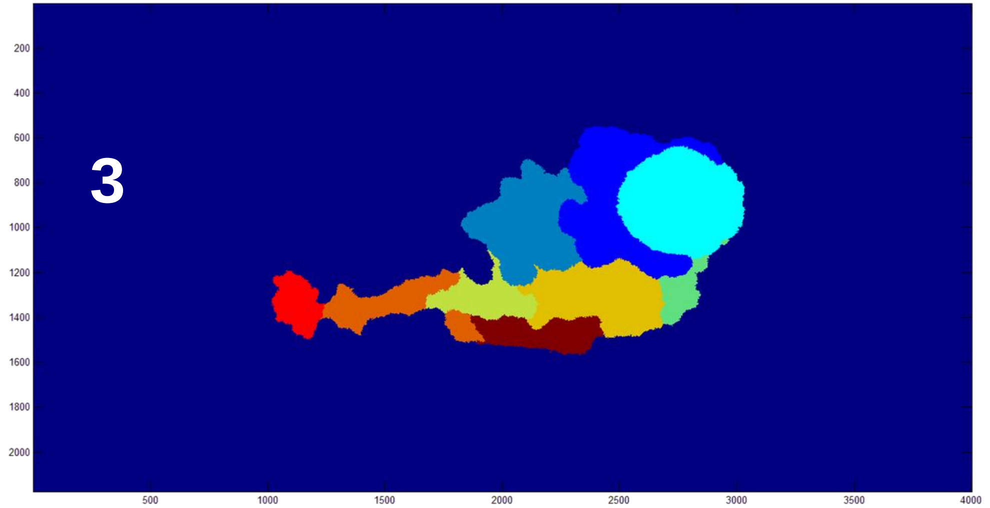
Abb 5.1. originale Österreichkarte

Dynamic Cartography - Population

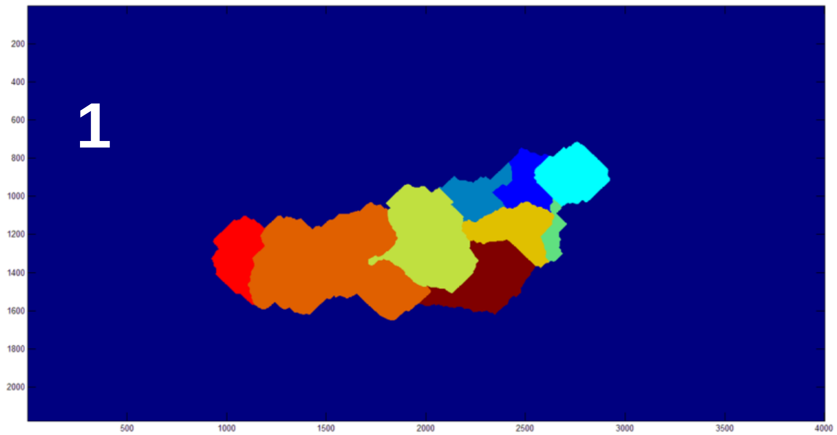
Mathematical Modelling and Simulation



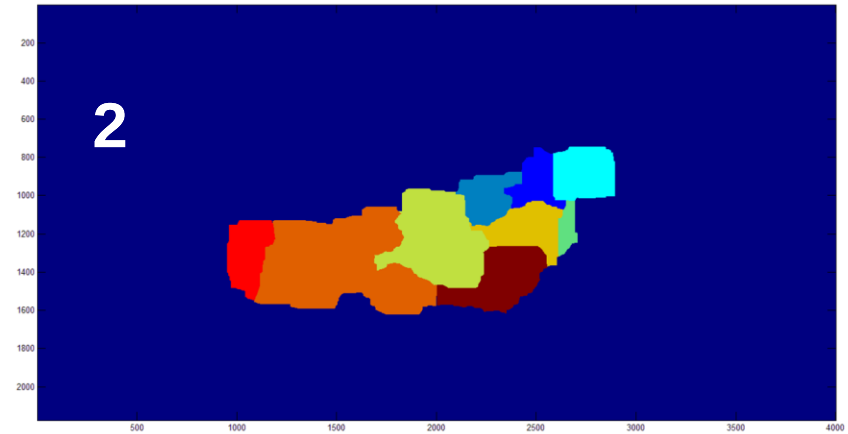




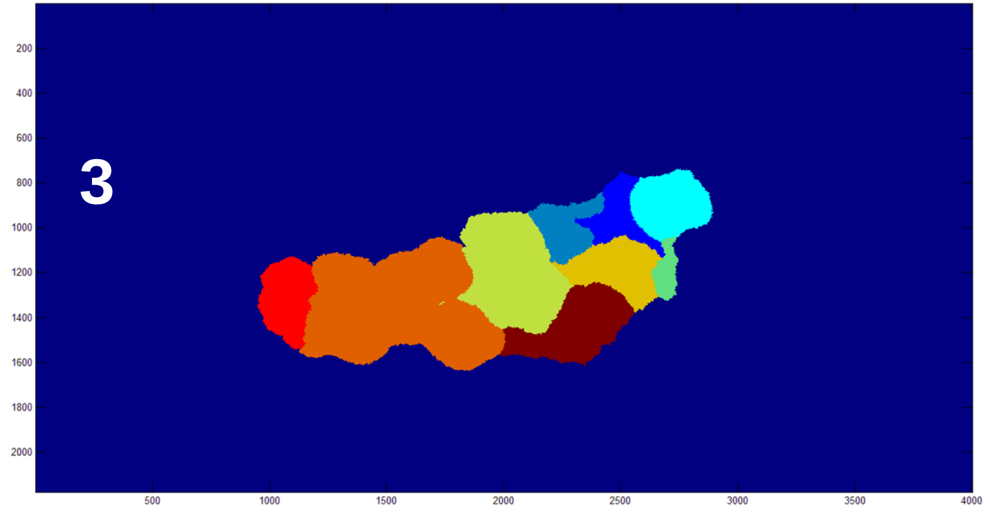
Dynamic Cartography - Tourism



Tourismus mit Neumann-Nachbarschaft



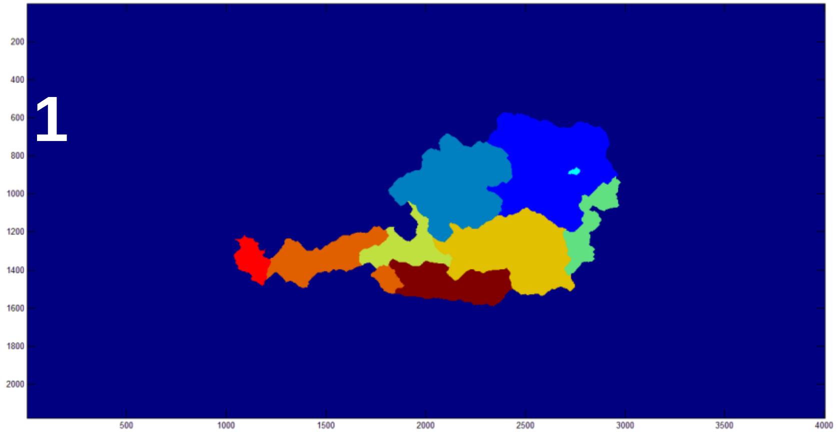
Tourismus mit Moore-Nachbarschaft



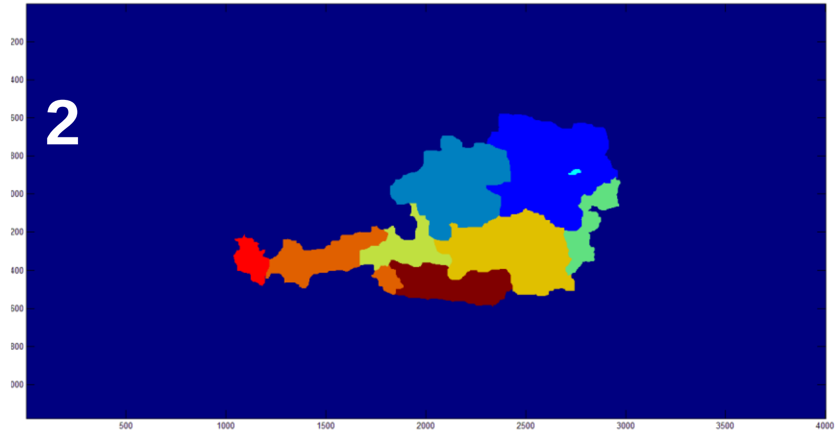
Tourismus mit "Wahrscheinlichkeitsschalter"

Dynamic Cartography – Hunting game

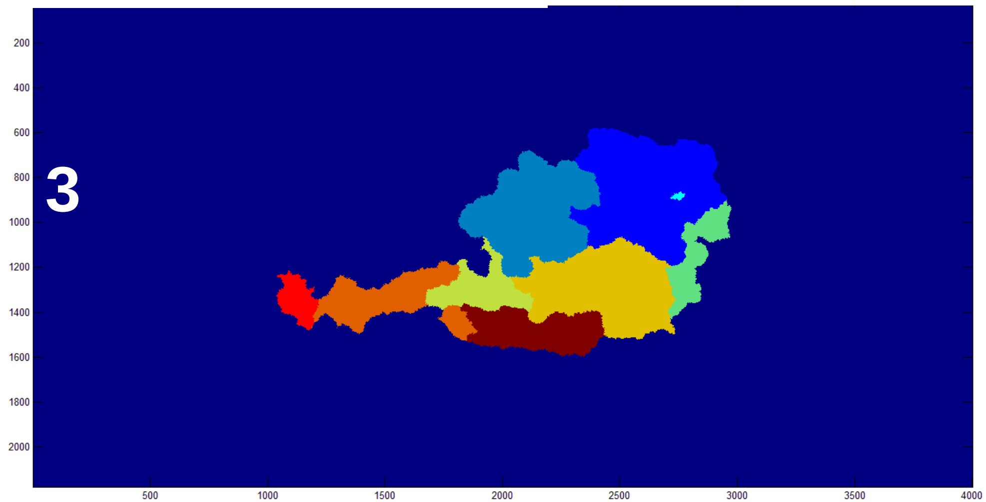
Mathematical Modelling and Simulation



Haarwildjagd mit Neumann-Nachbarschaft



Haarwildjagd mit Moore-Nachbarschaft



Haarwildjagd mit "Wahrscheinlichkeitsschalter"

Mathematical Modelling and Simulation

Example

LATTICE GAS CELLULAR AUTOMATA (LGCA)

Lattice Gas Cellular Automata

- Lattice Gas Cellular Automata (LGCA)
- Extension of the CA concept
- Intention: Simulate fluids and gases
- Invented by Hardy, Pomeau and de Pazzis (HPP automaton on square lattice), 1973
- Improved by Frisch, Hasslacher and Pomeau (FHP automaton on hexagonal grid), 1986

Lattice Gas Cellular Automata

Mathematical Modelling and Simulation

- **Ideas**
  - Cells do not have states but instead can contain particles
  - A particle can only proceed to a cell in the neighborhood
  - Instead of state updates, particles move to other cells
  - Particles represent the fluid or the gas

Lattice Gas Cellular Automata

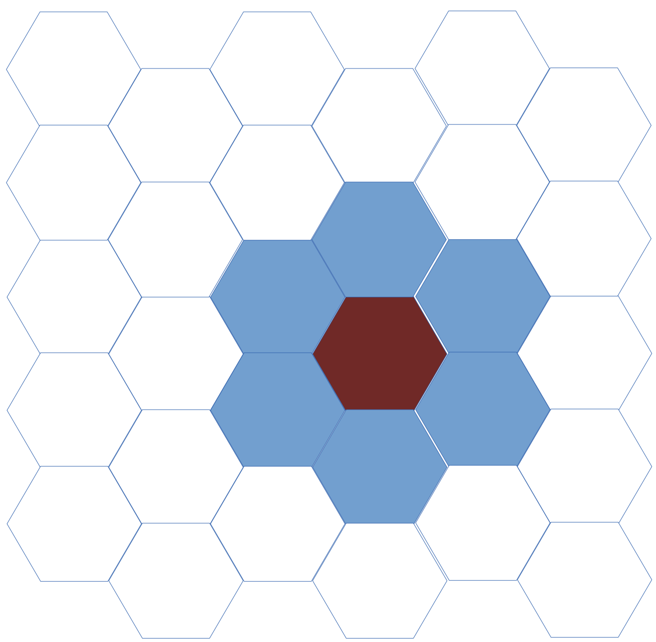
- **HPP**
  - square grid, Von-Neumann neighborhood, max. 4 particles per cell so that max. 1 particle goes to each neighbor
  - several issues when it comes to real interpretations (comparison with real fluids, validation)

Lattice Gas Cellular Automata

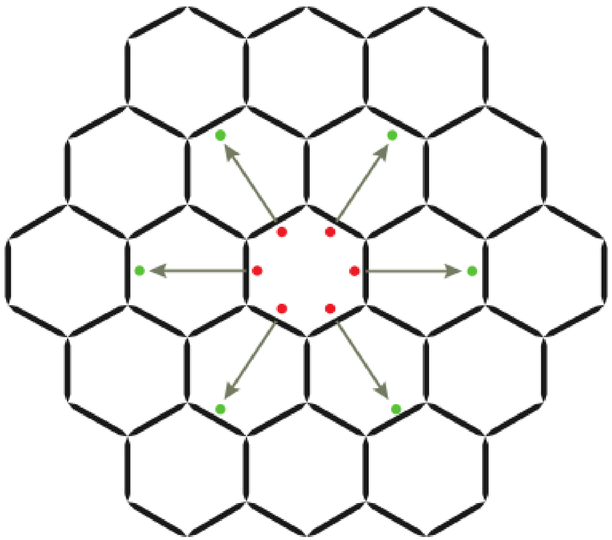
Mathematical Modelling and Simulation

- **FHP**
  - hexagonal grid
  - neighborhood = surrounding cells
  - max. 6 particles per cell, each going into a different direction → consistent definition
  - Corresponds to the Navier-Stokes-Equations → valid representation of fluid dynamics

FHP Model



neighbourhood



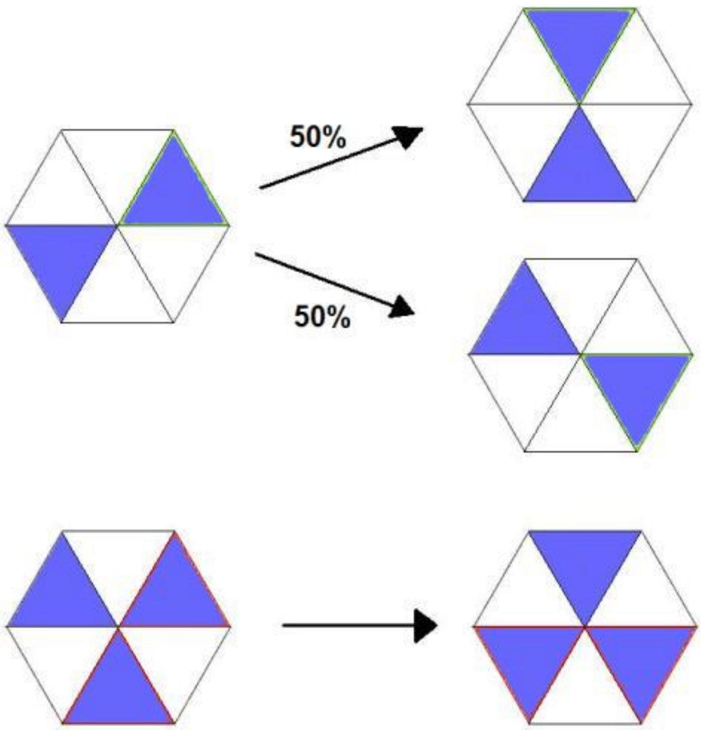
particles and directions

FHP Model

- Particle movements consist of two phases
  - Rotation of cells for special configurations
  - Movements of particles into their direction
- Developed by Wolf-Gladrow (2000)
- Different variations (FHP-I, FHP-II, FHP-III)

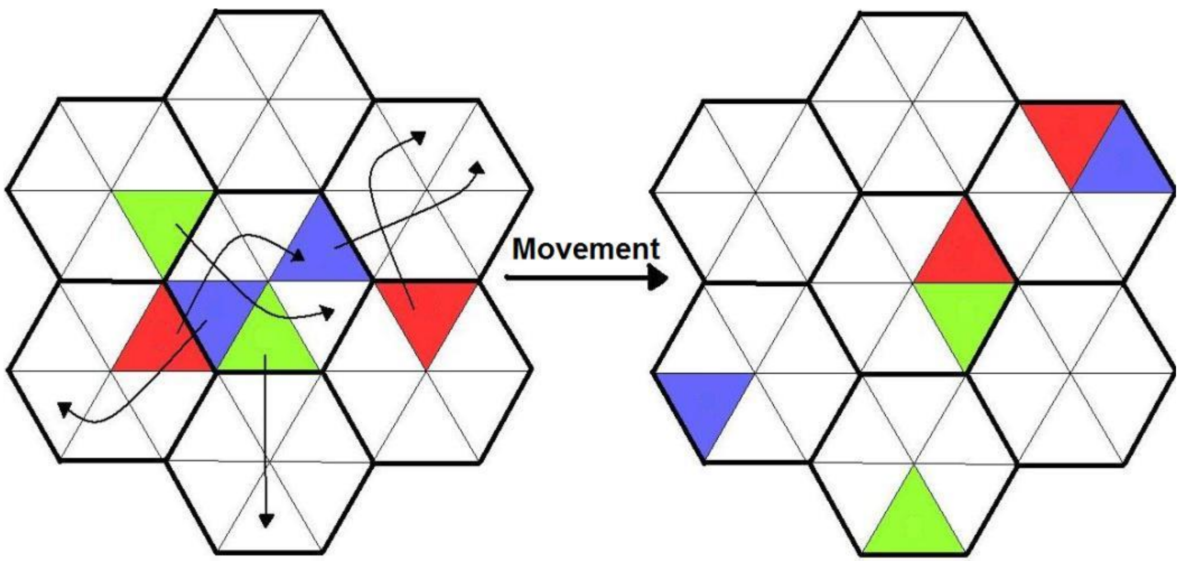
FHP Model

- Rotations
  - In the most simple case of FHP-I only for two situations
  - Provide a randomness



FHP Model

- Movements into designated directions



LGCAs

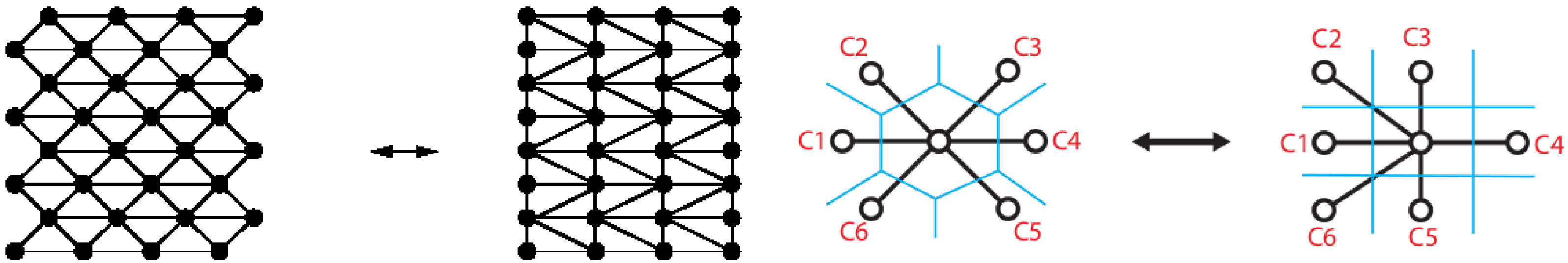
Mathematical Modelling and Simulation

- **Simulations & Visualizations**
- **HPP**
- http://en.wikipedia.org/wiki/File:Gas_velocity.gif
- **FHP**
- http://www.youtube.com/watch?v=HluQpDFOceg
- http://www.youtube.com/watch?v=00W6H7BGZ94

Remark: Implementation of a hexagonal grid

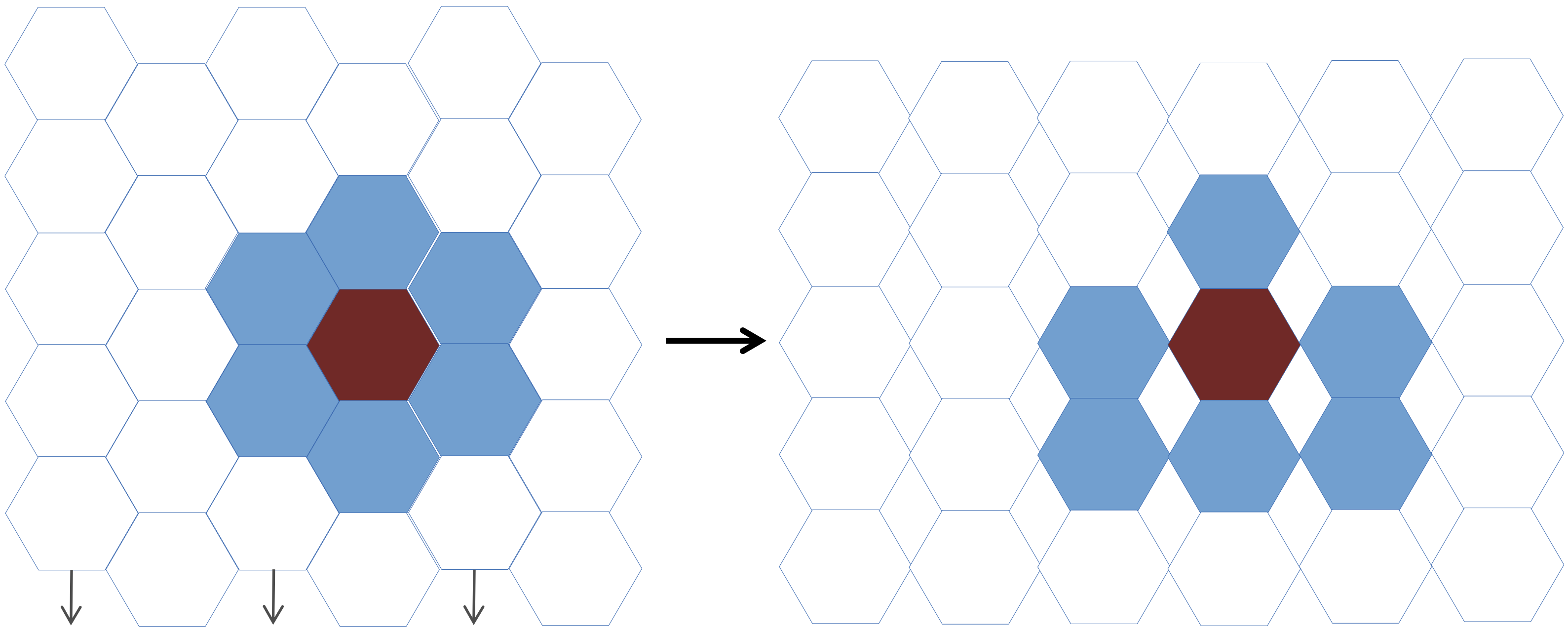
Mathematical Modelling and Simulation

- **Implementation**
  - hexagonally arranged grid → assign to a square lattice
  - conditional neighborhoods

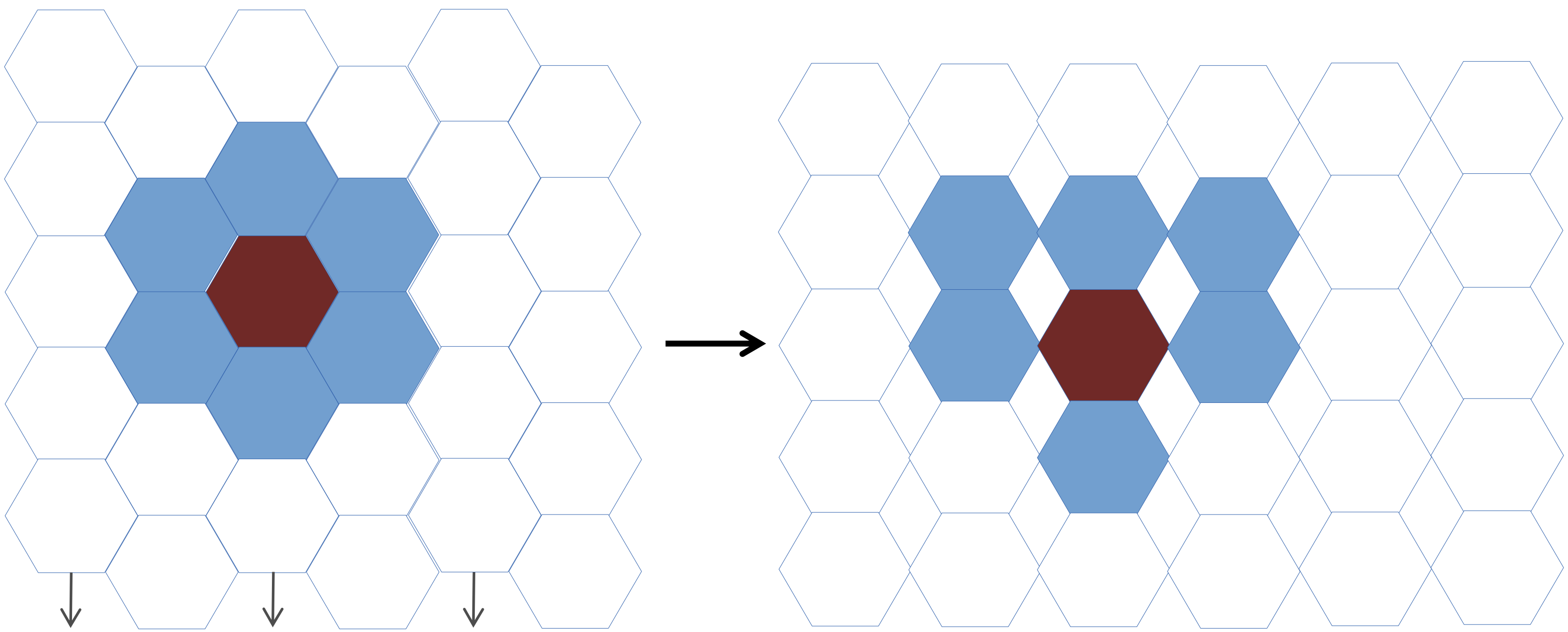


Remark: Implementation of a hexagonal grid

Mathematical Modelling and Simulation



Remark: Implementation of a hexagonal grid



Mathematical Modelling and Simulation

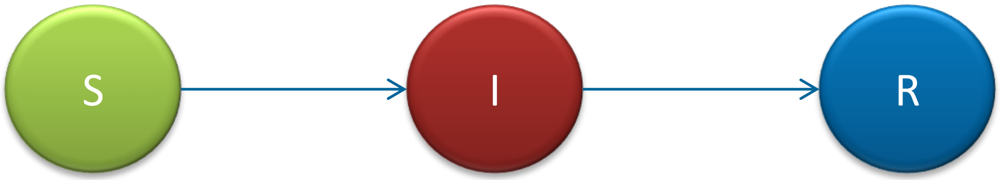
Example

EPIDEMIC SIMULATION WITH CA AND LGCA

SIR concept

Mathematical Modelling and Simulation

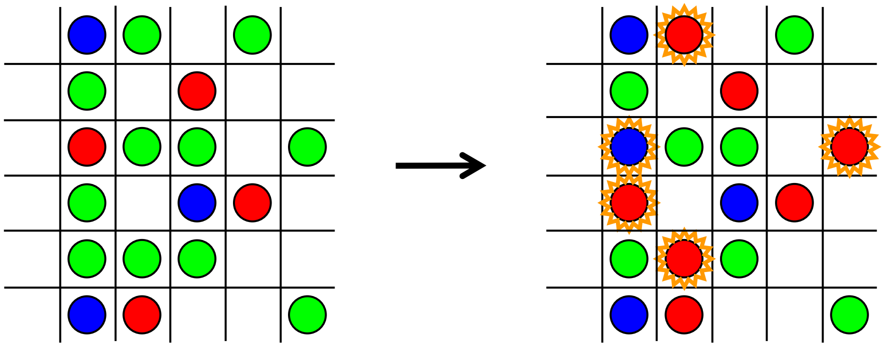
- Simulate the spread of an epidemics
- Susceptible (S) people become infected by infectous (I) and become resistant/recovered (R) after some time.
- Resistant persons cannot be infected again.



CA Implementation

CA Implementation of SIR epidemics:

- Every cell in a rectangular (hexagonal..) lattice represents a person/group of persons/household/…
- Infecious cells recover after some time (with some probability).
- Infectious cells may spread the disease to their neighbours (e.g. Moore neighbourhood)

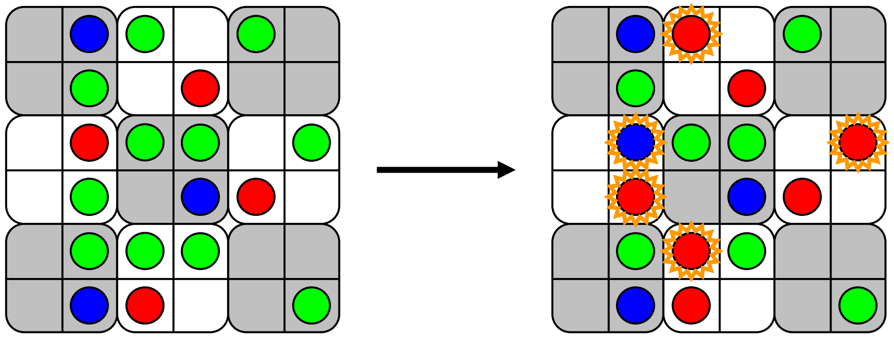


LGCA Implementation

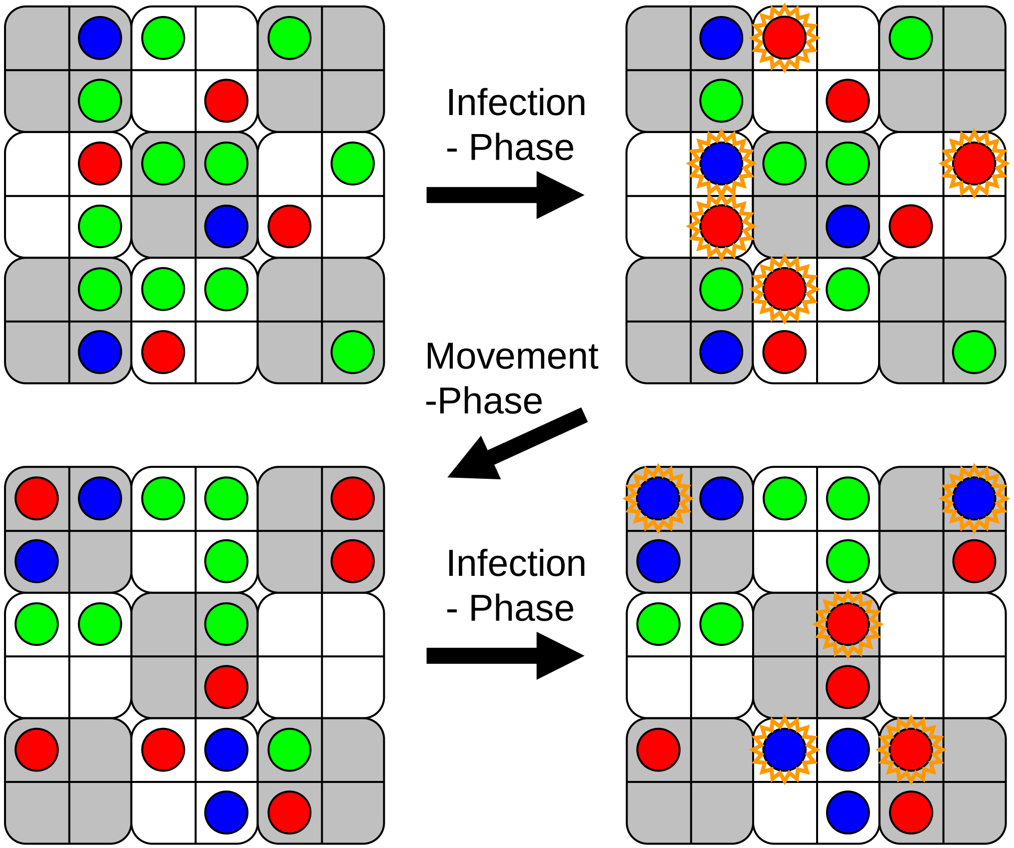
Mathematical Modelling and Simulation

LGCA Implementation of SIR epidemics:

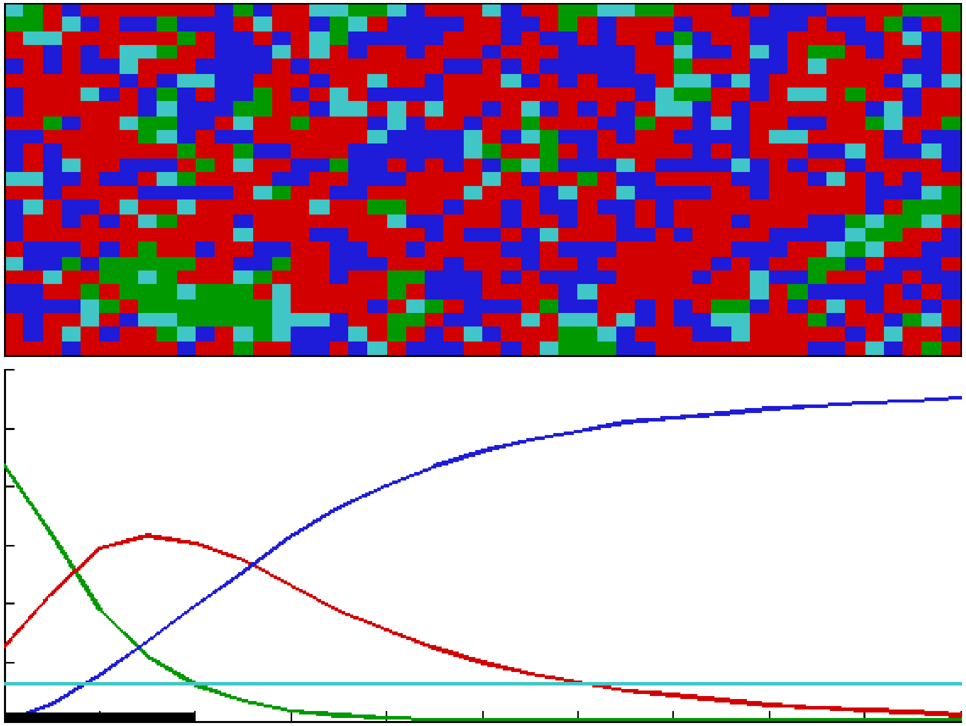
- Every cell in a rectangular (hexagonal..) lattice contains a number of persons (e.g. 4)
- Infecious persons recover after some time (with some probability).
- Infectious persons may spread the disease to all other persons in the cell



LGCA Implementation of SIR epidemics:

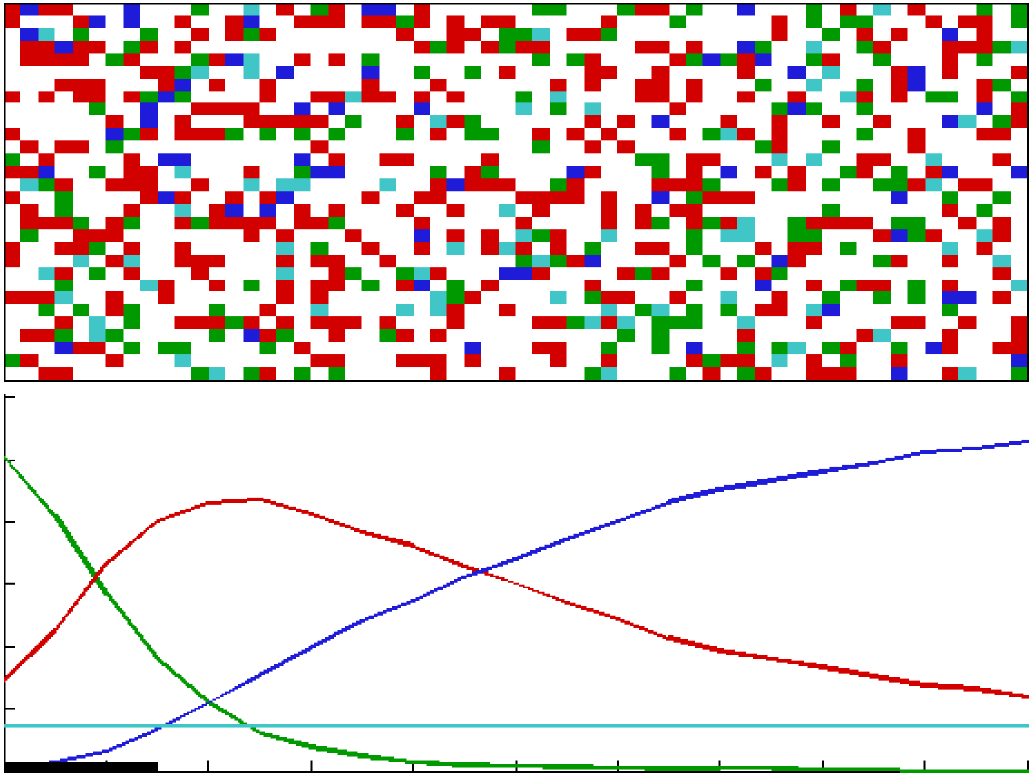


Epidemic simulation with CA



Epidemic simulation with HPP-LGCA

TU WIEN Mathematical Modelling and Simulation



HISTORY OF CELLULAR AUTOMATA

History of Cellular Automata

- 1925: Ising Modell
  - ferromagnetism, discrete model
- 1950: Von Neumann, Ulam
  - term “cellular automaton”
  - self reproductive, Von-Neumanns theory on logic automata

- 1950-1970: Zuse, u.a.
  - parallel algorithms
  - discrete processes (e.g. PDEs)
- 1970s: Hardy, Pomeau, de Pazzis
  - Lattice Gas Cellular Automata
- 1979: Conway's Game of Life

History of Cellular Automata

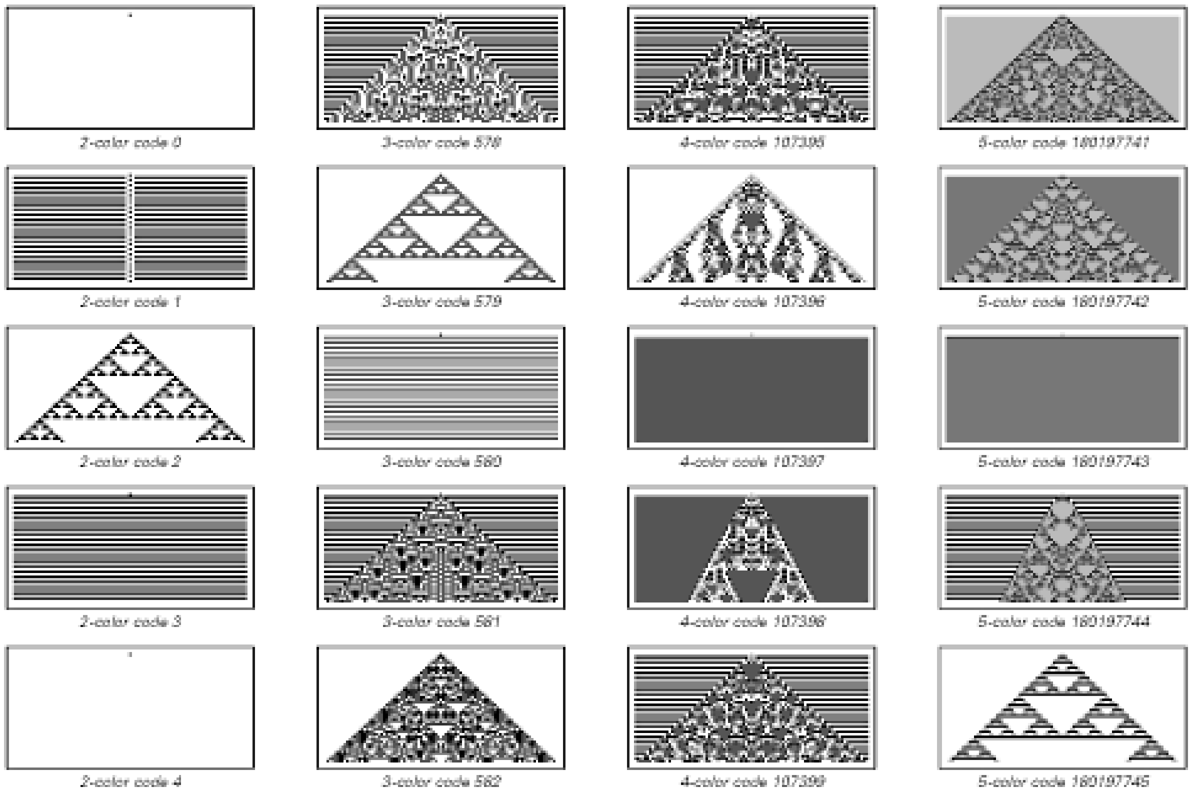
Mathematical Modelling and Simulation

- 1980+: different applications
- 2002: Wolfram
  - complete classifications of 1-dimensional cellular automata

History of Cellular Automata

Mathematical Modelling and Simulation

Stephen Wolfram, „A new Kind of Science“



History of Cellular Automata

Mathematical Modelling and Simulation

- a spatially extended decentralized system made up of a number of individual components […] local interaction […] depending on the states of its local neighbors […] parallel processing [Ganguly]
- regular grid of cells, each in one of a finite number of states […] neighbourhood […] new generation is created according to some fixed rule [Wikipedia]

History of Cellular Automata

Mathematical Modelling and Simulation

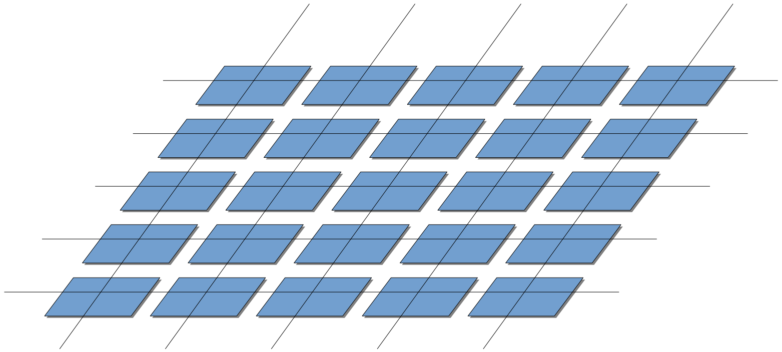
- regular arrangements of single cells [...] each cell holds a finite number of discrete states [...] updated simultaneously [...] the rules for the evolution of a cell depend only on a local neighborhood [Gladrow]

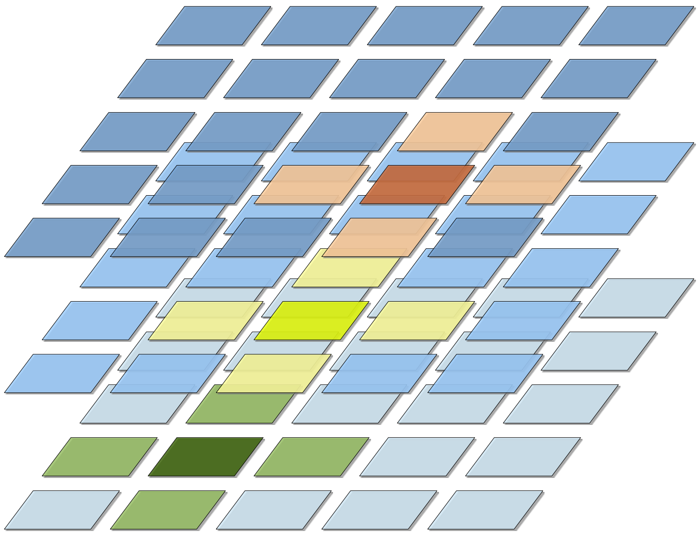
5. Conclusions

CONCLUSIONS

Characteristics of CA

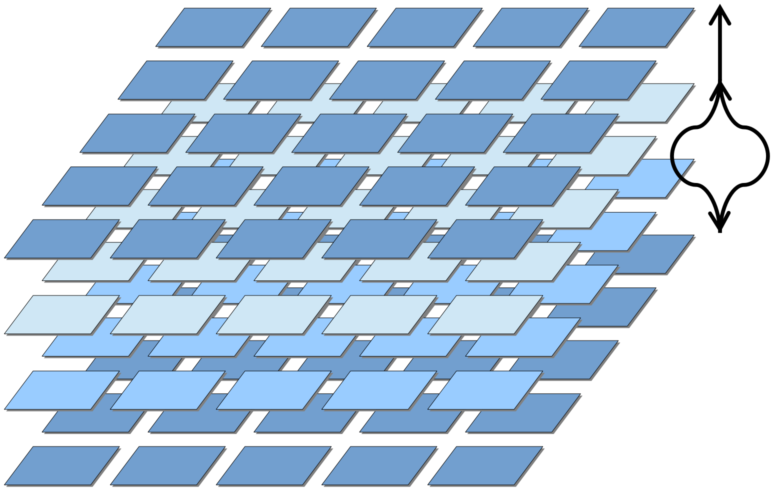
- Regular lattice, same kind of neighbourhoods





Characteristics of CA

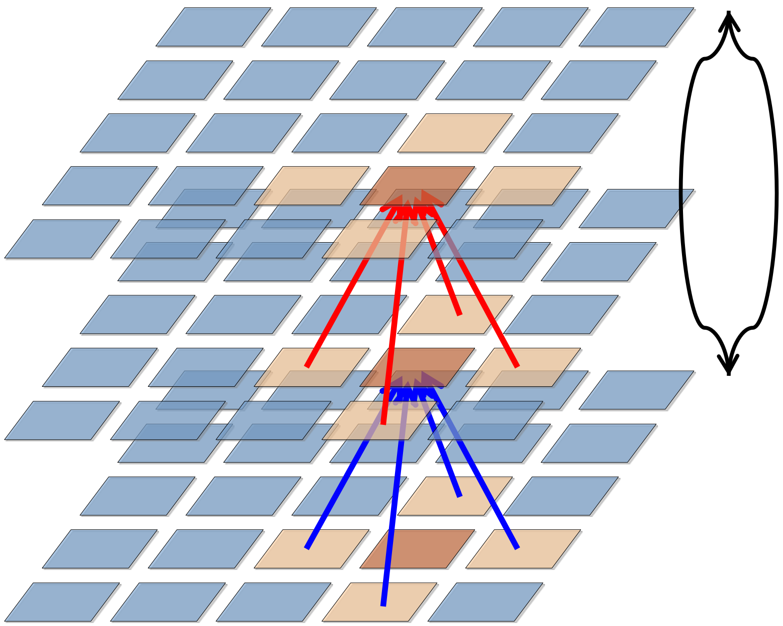
- Discrete time, equidistant time steps



Characteristics of CA

Mathematical Modelling and Simulation

- Spatial representation, locality

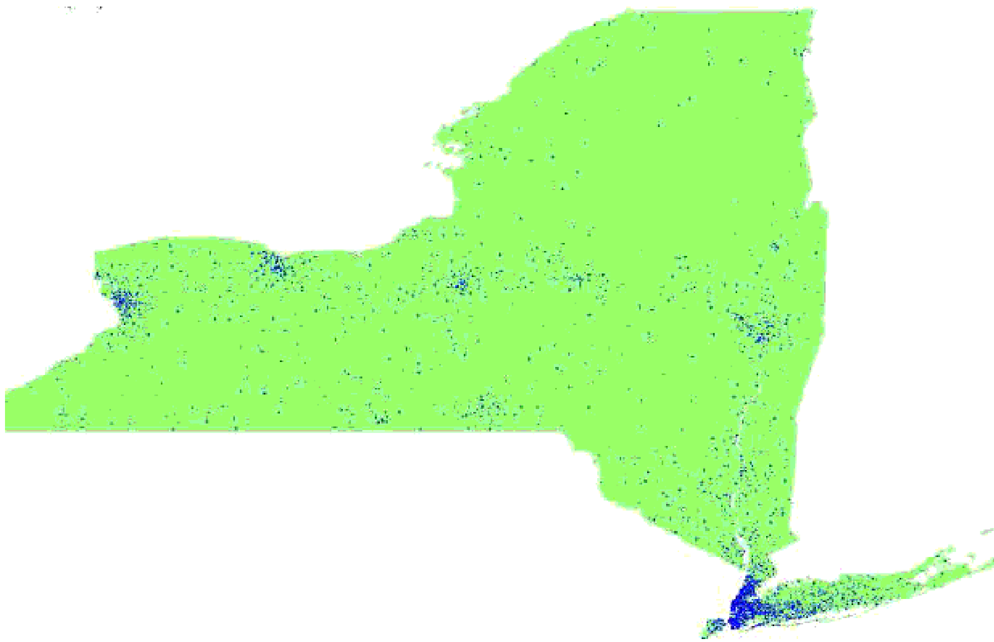


Applications for Cellular Automata

Dynamic Cartography

Mathematical Modelling and Simulation

- map shows relation between sizes
- The dots symbolises cancer patients



Dynamic Cartography

Mathematical Modelling and Simulation

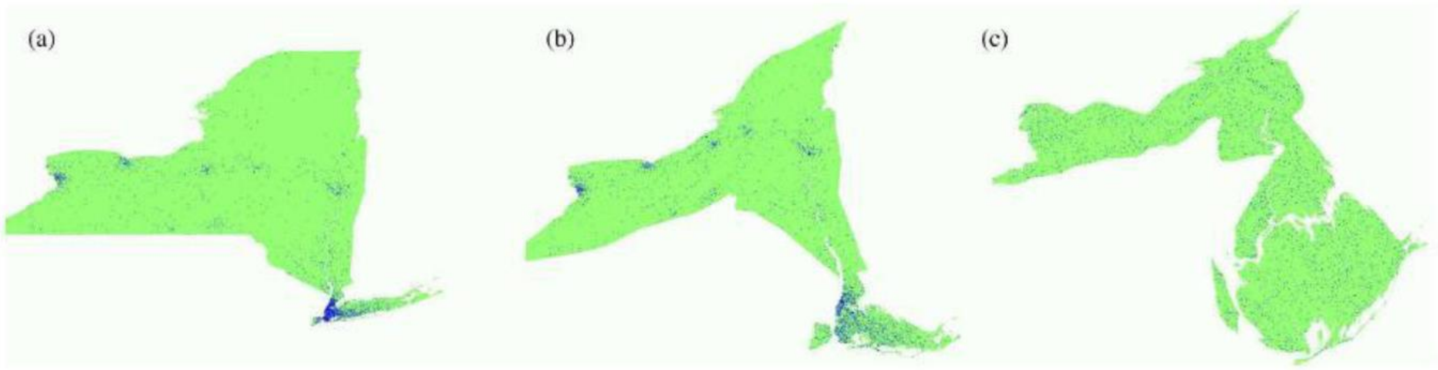


Figure 2: Visualization of lung cancer cases among males in the state of New York 1993–1997. Each dot represents ten cases, randomly placed within the zip-code area of occurrence. (a) The original map. (b) A cartogram using a coarse-grained population density with $\sigma = 0.3°$. (c) A cartogram using a much finer-grained population density ($\sigma = 0.04°$). (Data from the New York State Department of Health.)

Dynamic Cartography

- Amount of cancer patients spread equally to squares in each region (e.g. staats)
- Diffusion from places with high density to low
- Diffusion continues until the density is equal distributed
- Regions with higher density grow, others shrink

Dynamic Cartography

Mathematical Modelling and Simulation

Neumann model

1,5	1,5	2	1,5
1,5	2	2	1,5
1,5	1,5	2	2
1,5	1,5	2	1,5

5	0,9	0,9	0,9
0,9	0,9	0,9	0,9
5	0,9	0,9	0,9
5	0,9	0,9	0,9

Density depending model

1,5	1,5	2	1,5
1,5	2	2	1,5
1,5	1,5	2	2
1,5	1,5	2	1,5

2,5	1,2	1,2	2,5
1,2	1,2	1,2	1,2
2,5	1,2	1,2	1,2
2,5	2,5	1,2	2,5

Moore model

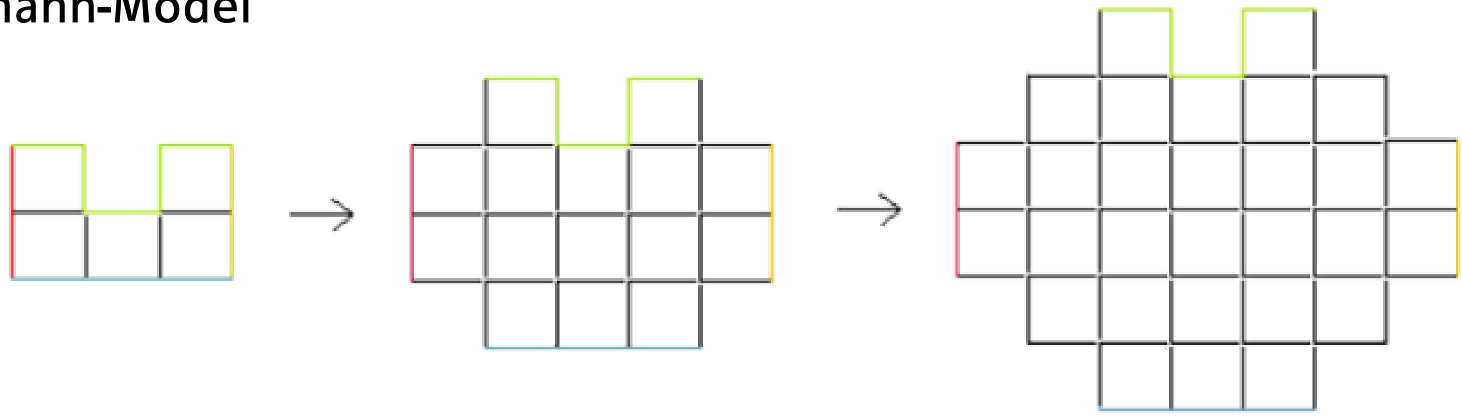
1,5	1,5	2	1,5
1,5	2	2	1,5
1,5	1,5	2	2
1,5	1,5	2	1,5

0,8	0,8	0,8	0,8
0,8	0,8	0,8	0,8
0,8	0,8	0,8	0,8
5	0,8	0,8	0,8

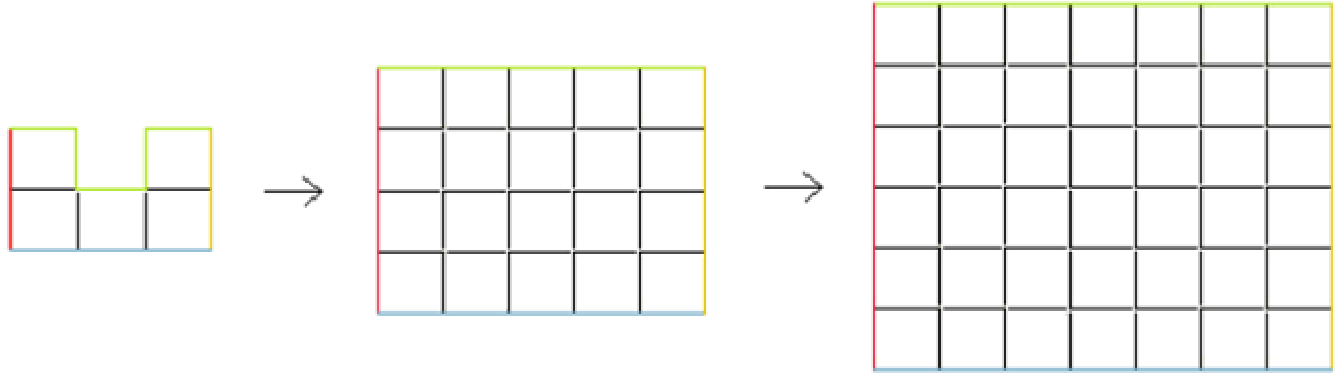
Dynamic Cartography - Boundaries

Mathematical Modelling and Simulation

Neumann-Model



Moore-Model



Dynamic Cartography - Results

Mathematical Modelling and Simulation

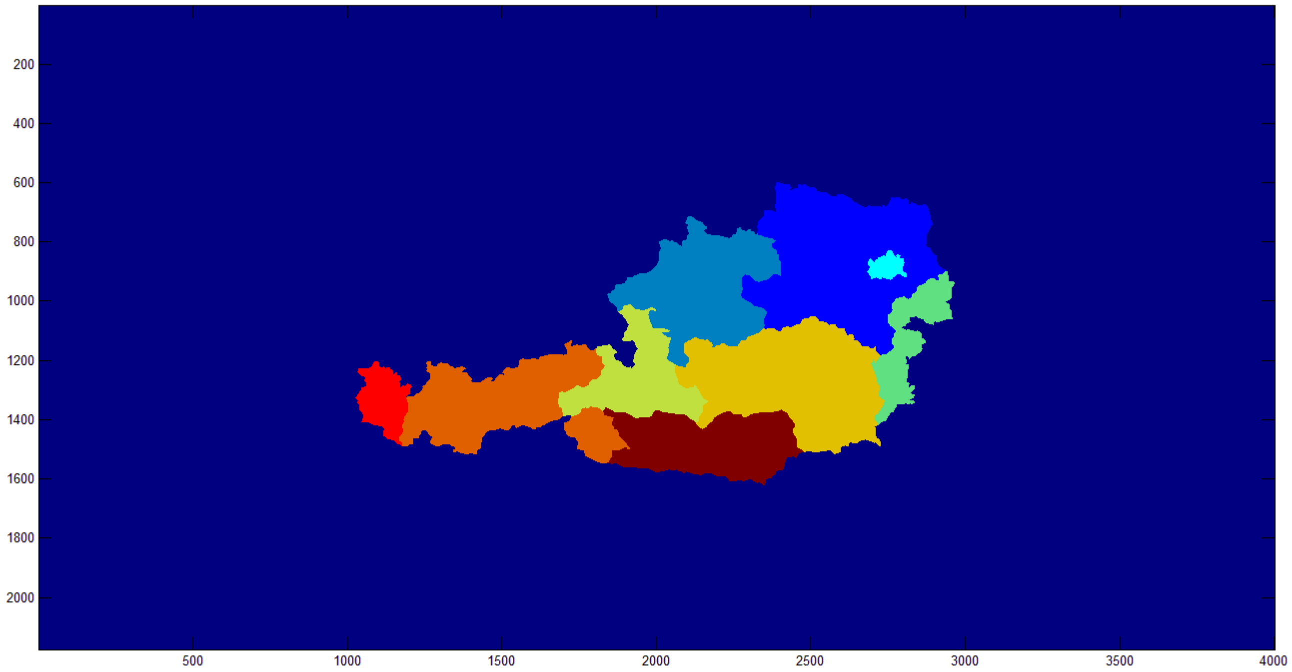
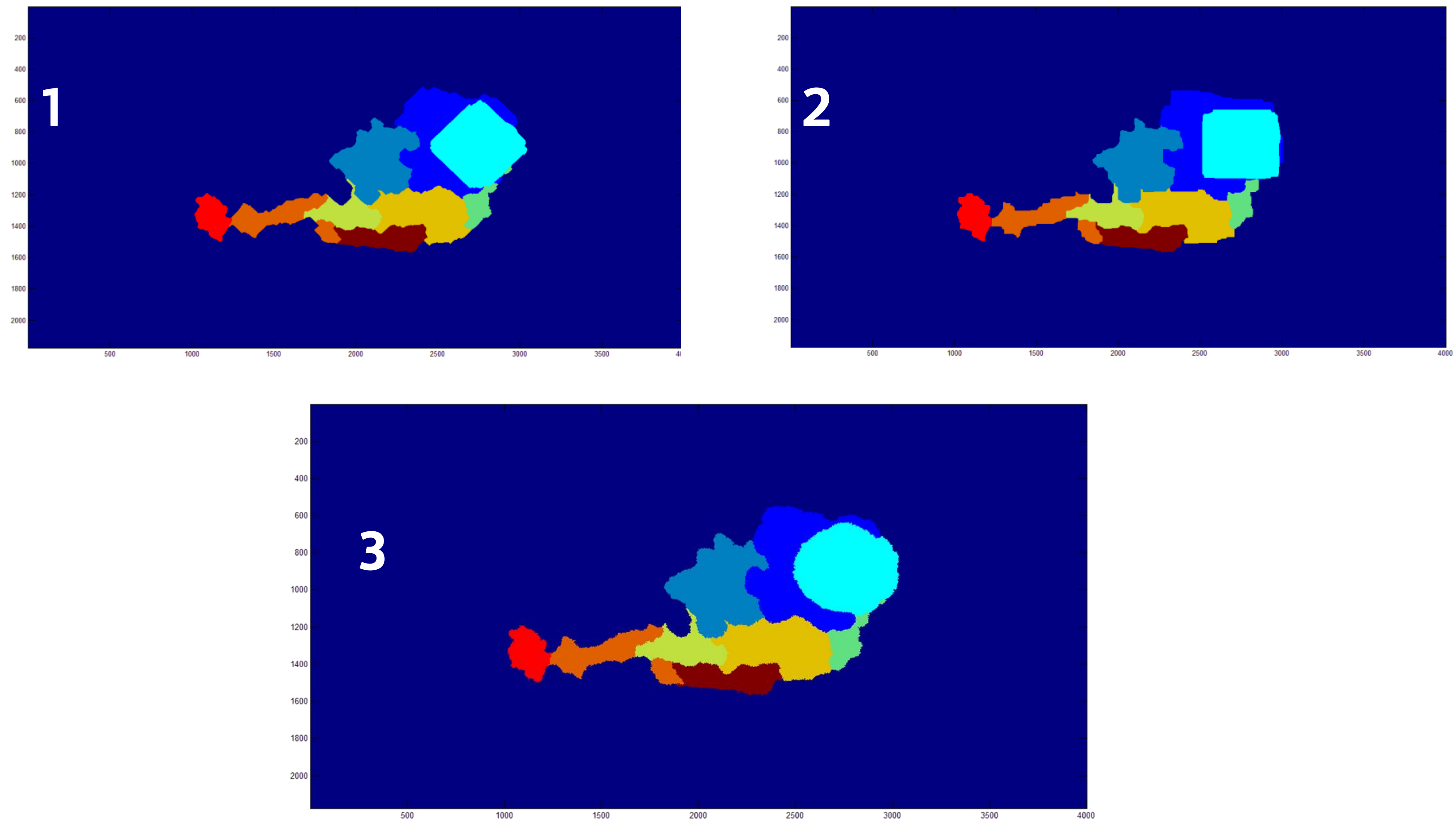


Abb 5.1. originale Österreichkarte

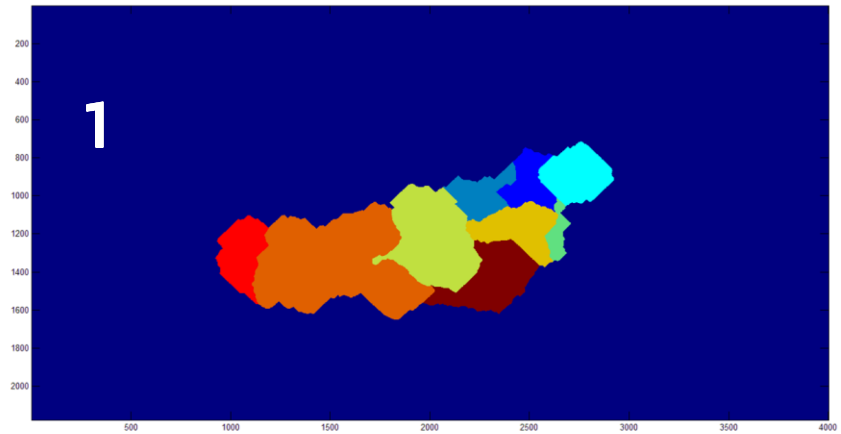
Dynamic Cartography - Population

TU WIEN Mathematical Modelling and Simulation

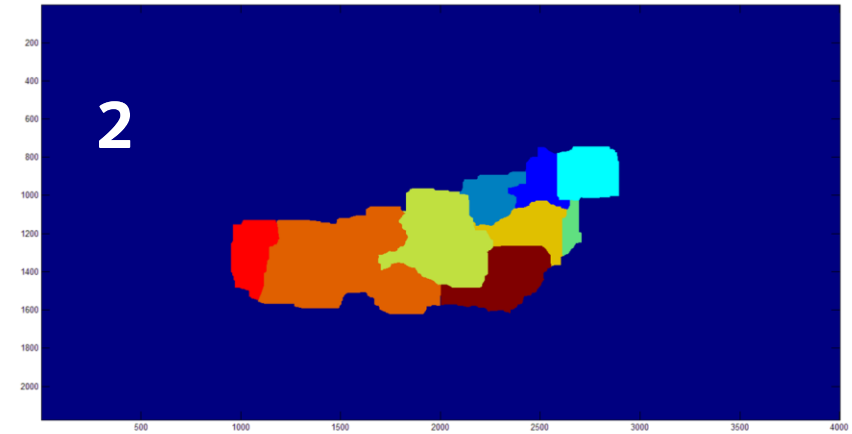


Dynamic Cartography - Tourism

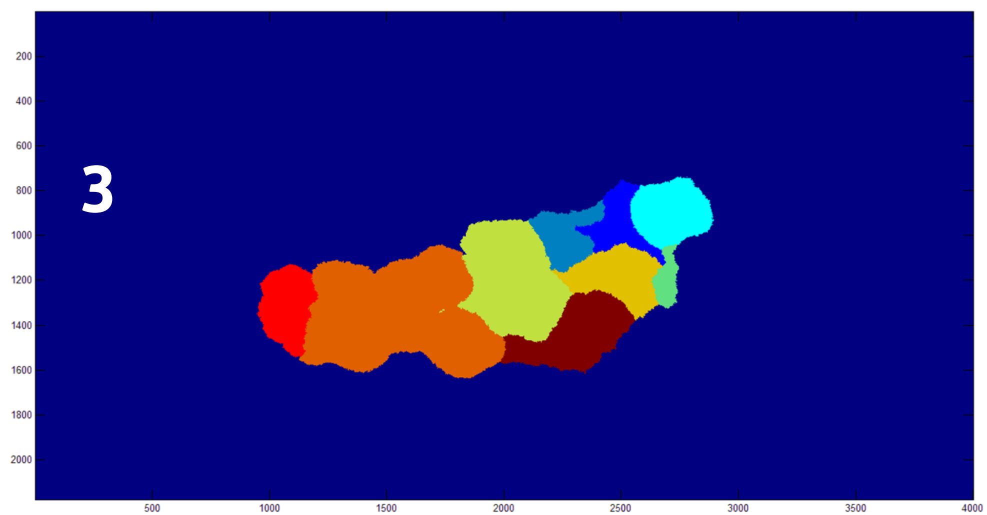
Mathematical Modelling and Simulation



Tourismus mit Neumann-Nachbarschaft

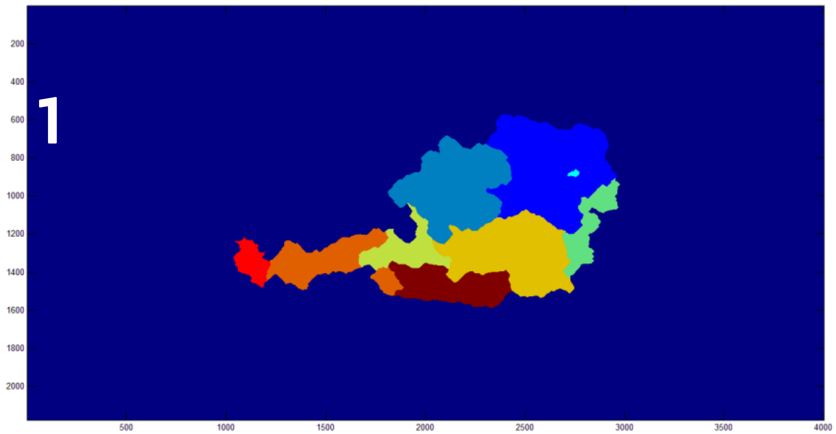


Tourismus mit Moore-Nachbarschaft

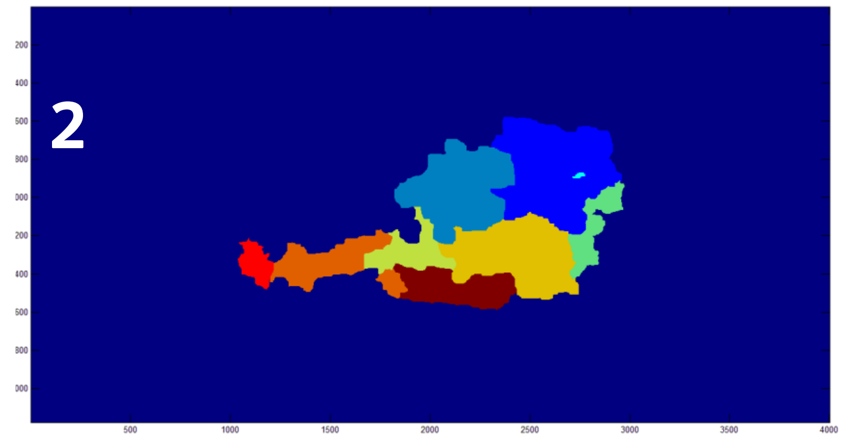


Tourismus mit "Wahrscheinlichkeitsschalter"

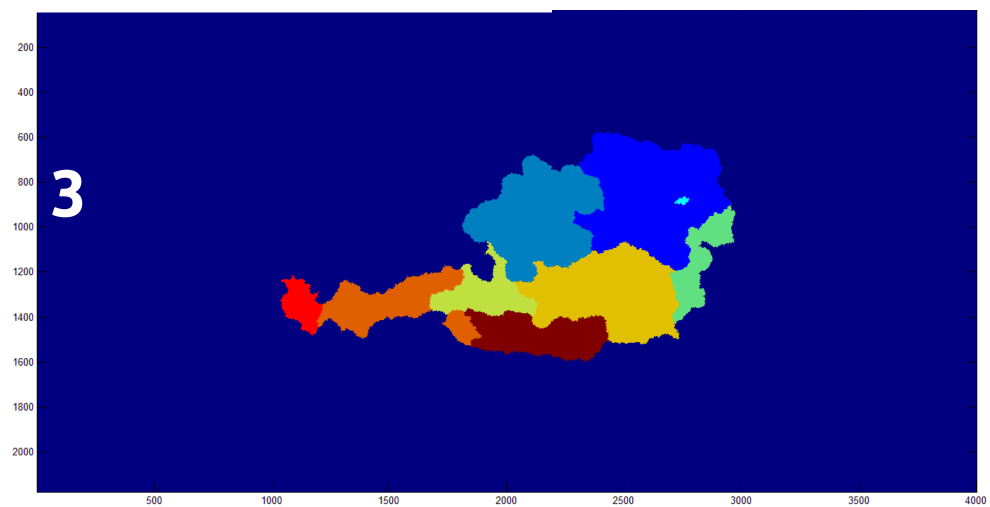
Dynamic Cartography – Hunting game



Haarwildjagd mit Neumann-Nachbarschaft



Haarwildjagd mit Moore-Nachbarschaft



Haarwildjagd mit "Wahrscheinlichkeitsschalter"

Application Example: Traffic Simulation

Mathematical Modelling and Simulation

Nagel-Schreckenberg-Model

- discretisation of a road or motorway into cells of approximately 4m
- possible states:
  - $s = 0$: no vehicle
  - $s > 0$: speed of vehicle
- update rules (implicitly defined!):
  - accelerate: **IF** $v < v\_\max$ **AND** next vehicle $v+1$ cells away **THEN** $v(t+1) = v(t) + 1$
  - brake: **IF** next vehicle j cells away **AND** $j < v$ **THEN** $v(t+1) = j - 1$
  - randomisation: $v(t+1) = v(t) - 1$ **with a certain probability**
  - movement: $s(t+1) = s(t) + v(t)$

Application Example: Traffic Simulation

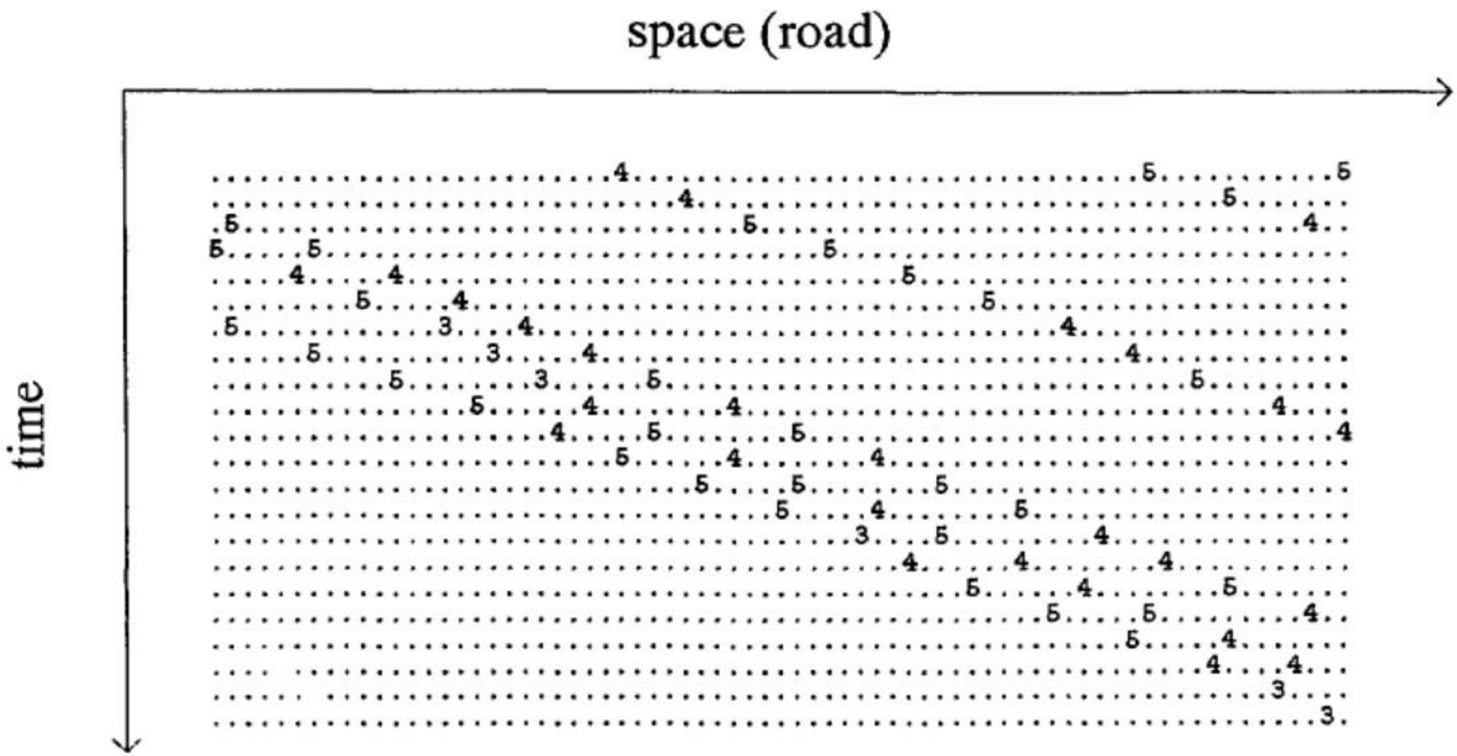


Fig.1. — Simulated traffic at a (low) density of 0.03 cars per site. Each new line shows the traffic lane after one further complete velocity-update and just before the car motion. Empty sites are represented by a dot, sites which are occupied by a car are represented by the integer number of its velocity. At low densities, we see undisturbed motion.

Application Example: Traffic Simulation

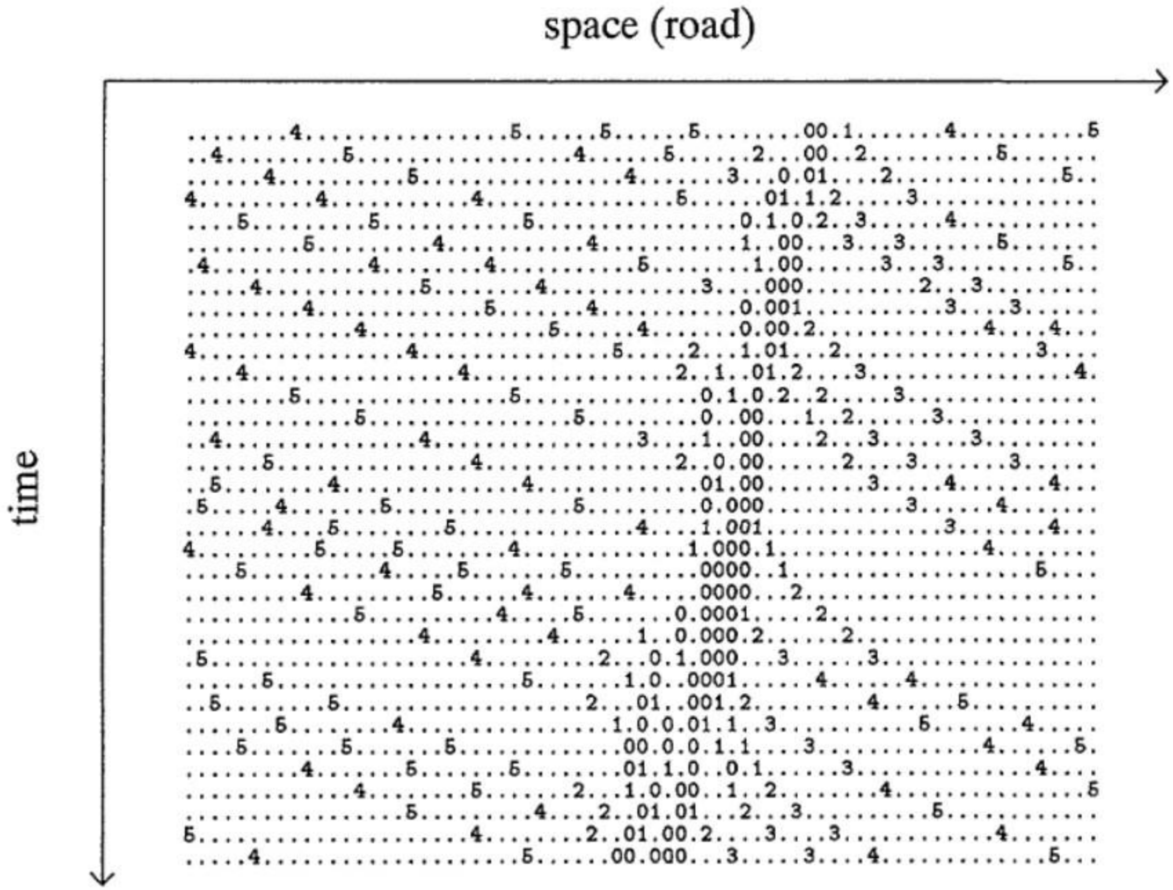


Fig.2. — Same picture as figure 1, but at a higher density of 0.1 cars per site. Note the backward motion of the traffic jam.

Application Example: Traffic Simulation

Mathematical Modelling and Simulation

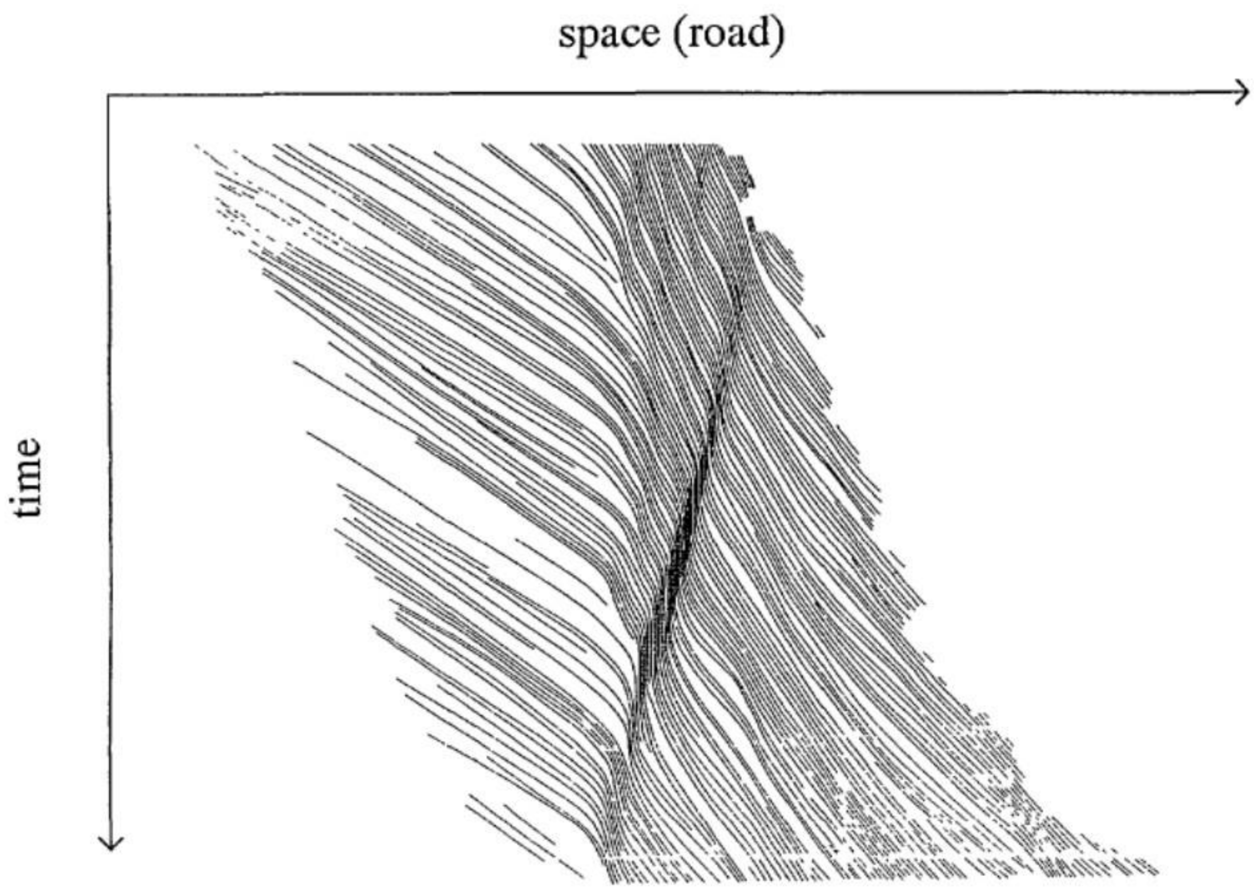


Fig.3. — Space-time-lines (trajectories) for cars from Aerial Photography (after [16]). Each line represents the movement of one vehicle in the space-time-domain.

Introduction to Agent-Based Modelling

Historical Background

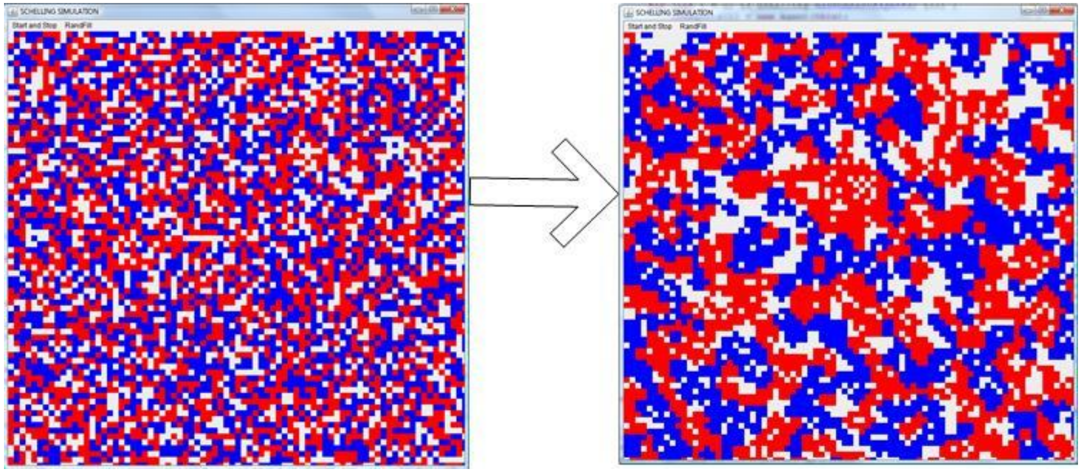
Mathematical Modelling and Simulation

- 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

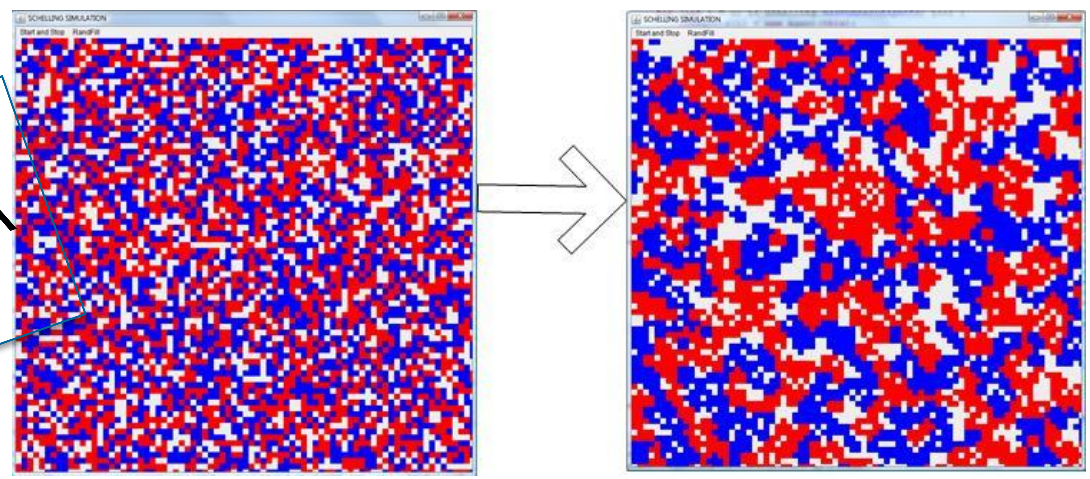
- Agent-based modelling is a comparably young modelling technique.
- Were inspired by Cellular Automata (Von Neumann, Ulam, etc)
- Thomas Schelling‘s Model of Segregation (1971) is broadly denoted as the first agent-based model



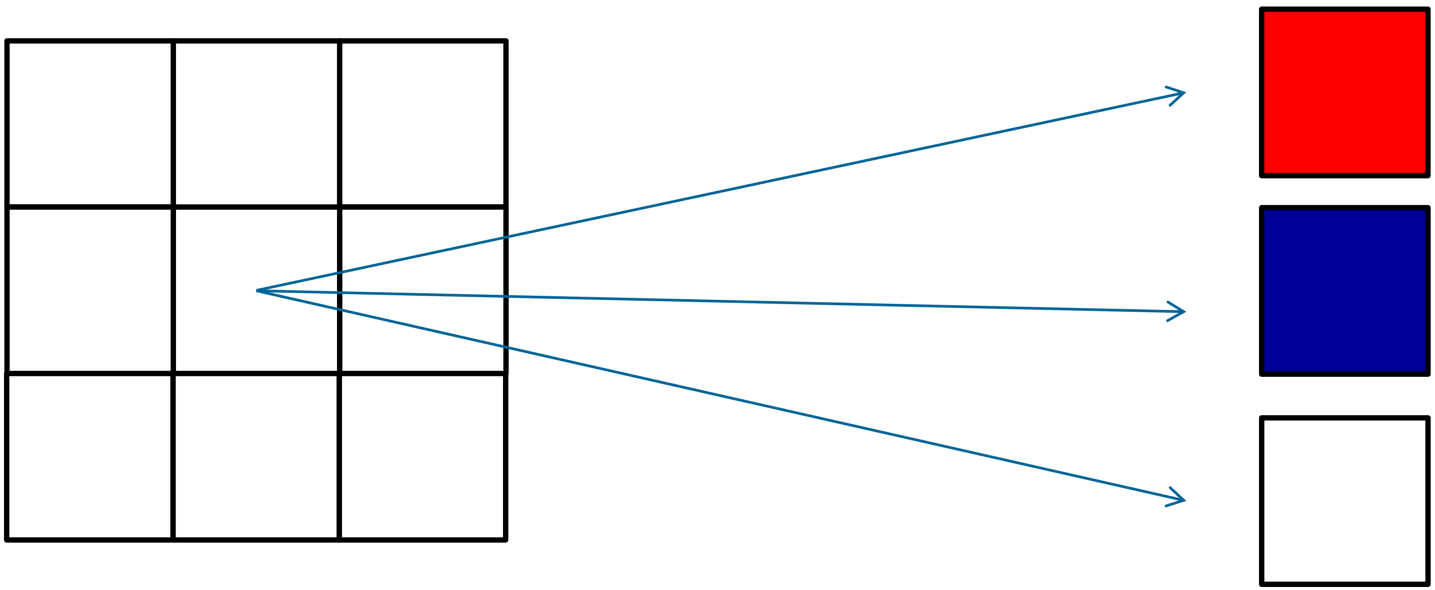
Historical Background

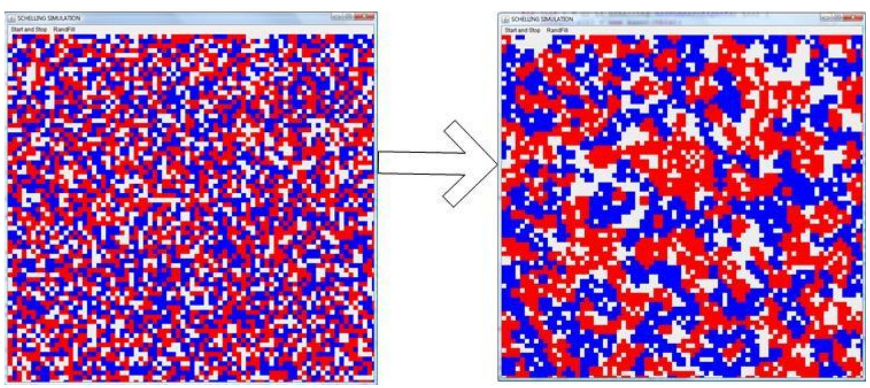
- Agent-based modelling is a comparably young modelling technique.
- Were inspired by Cellular Automata (Von Neumann, Ulam, etc)
- Thomas Schelling‘s Model of Segregation (1971) is broadly denoted as the first agent-based model

Still Looks very much like a CA model?!



A Small but Powerful Difference…

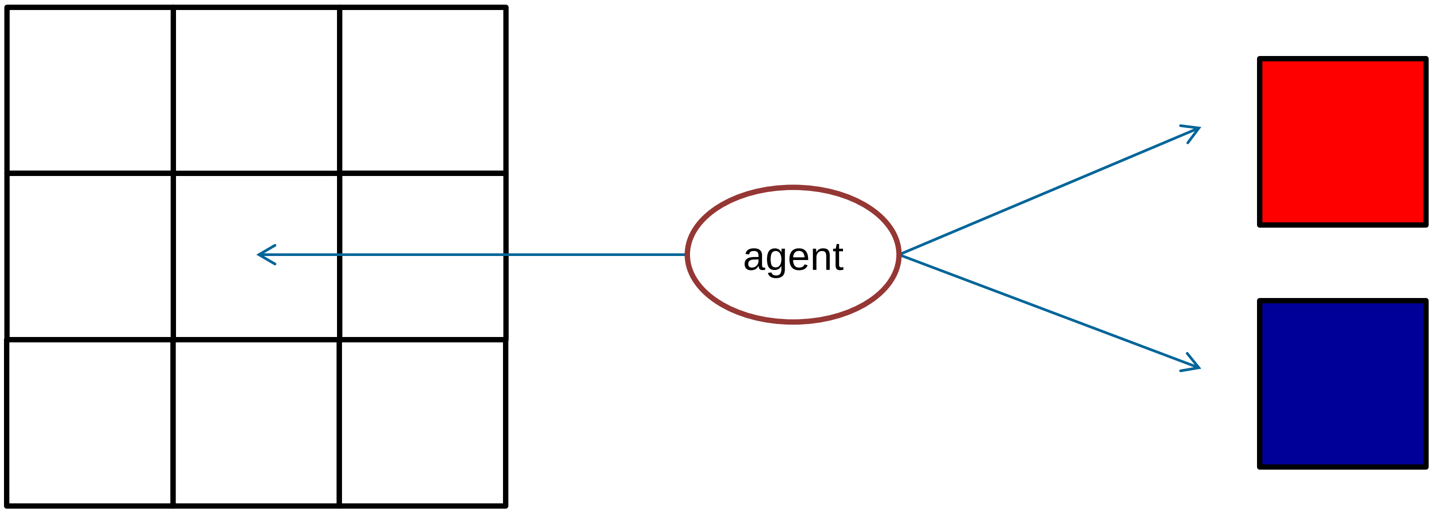


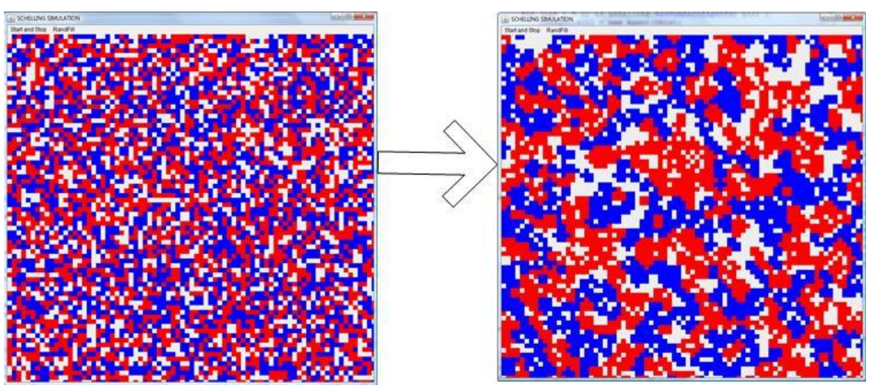


CA Model

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…

Mathematical Modelling and Simulation

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 introduce death of agents

We could add some immigrants

Agent

Why Agent?

Agent

Mathematical Modelling and Simulation

Latin: „agere“ (to act)

What is an Agent? (1)

Mathematical Modelling and Simulation

- 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? (2)

- 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? (2)

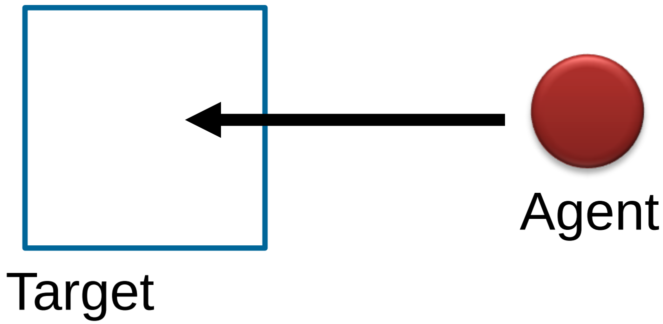
- With respect to Winter Simulation Conference (2005 & 2006) an agent has to...

  ... **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)

Agent

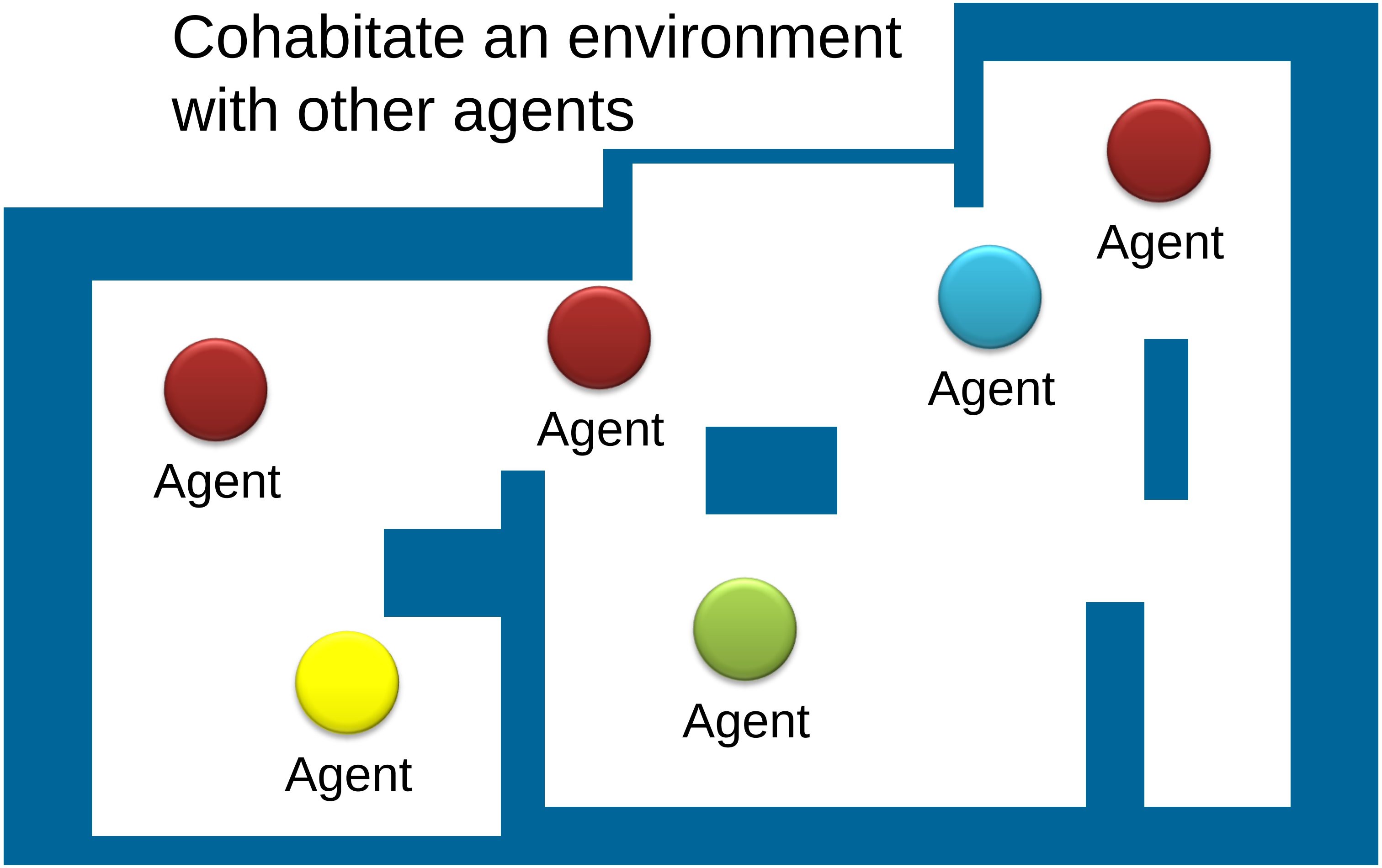
Act Targeted

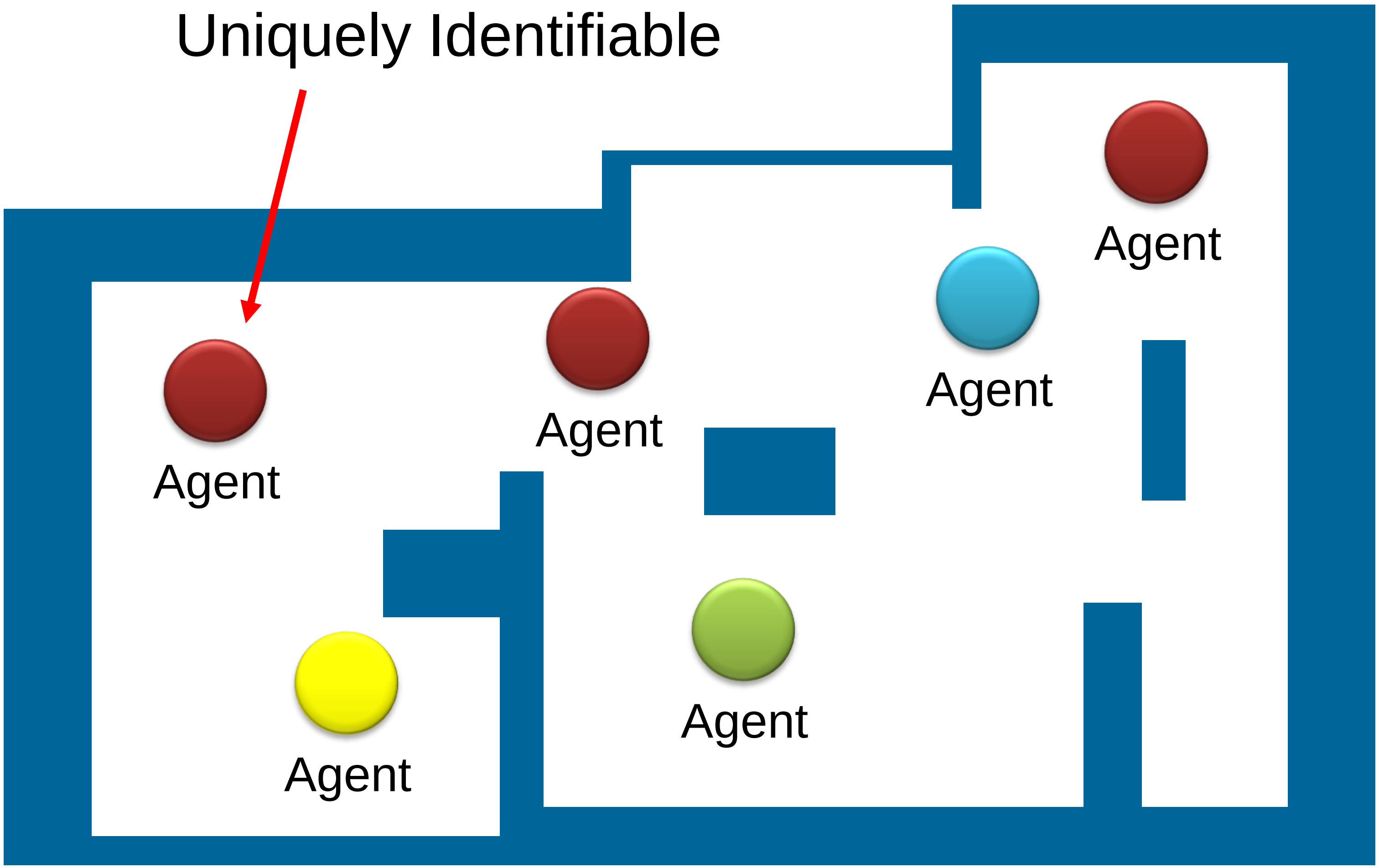


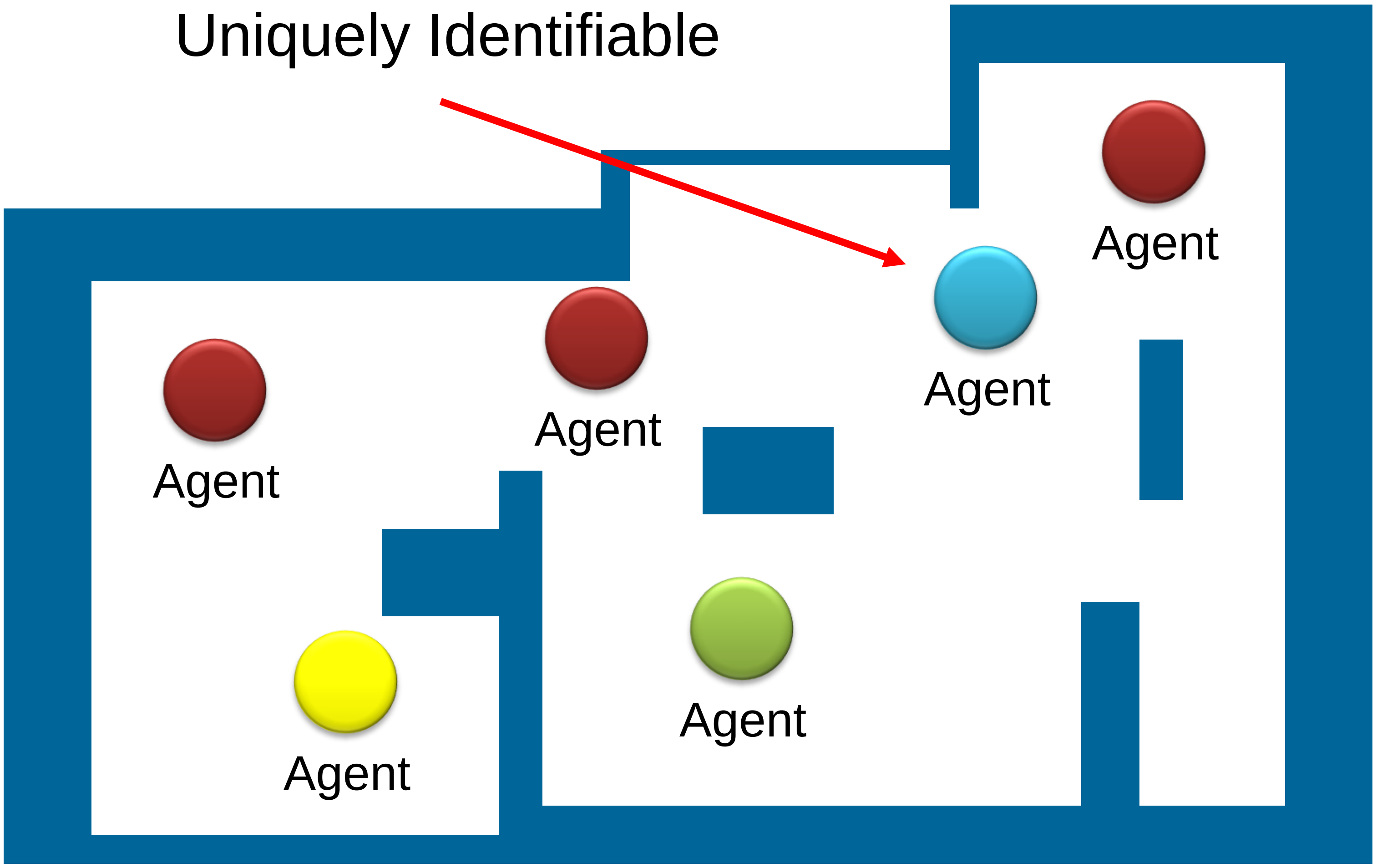
Act Targeted

Target

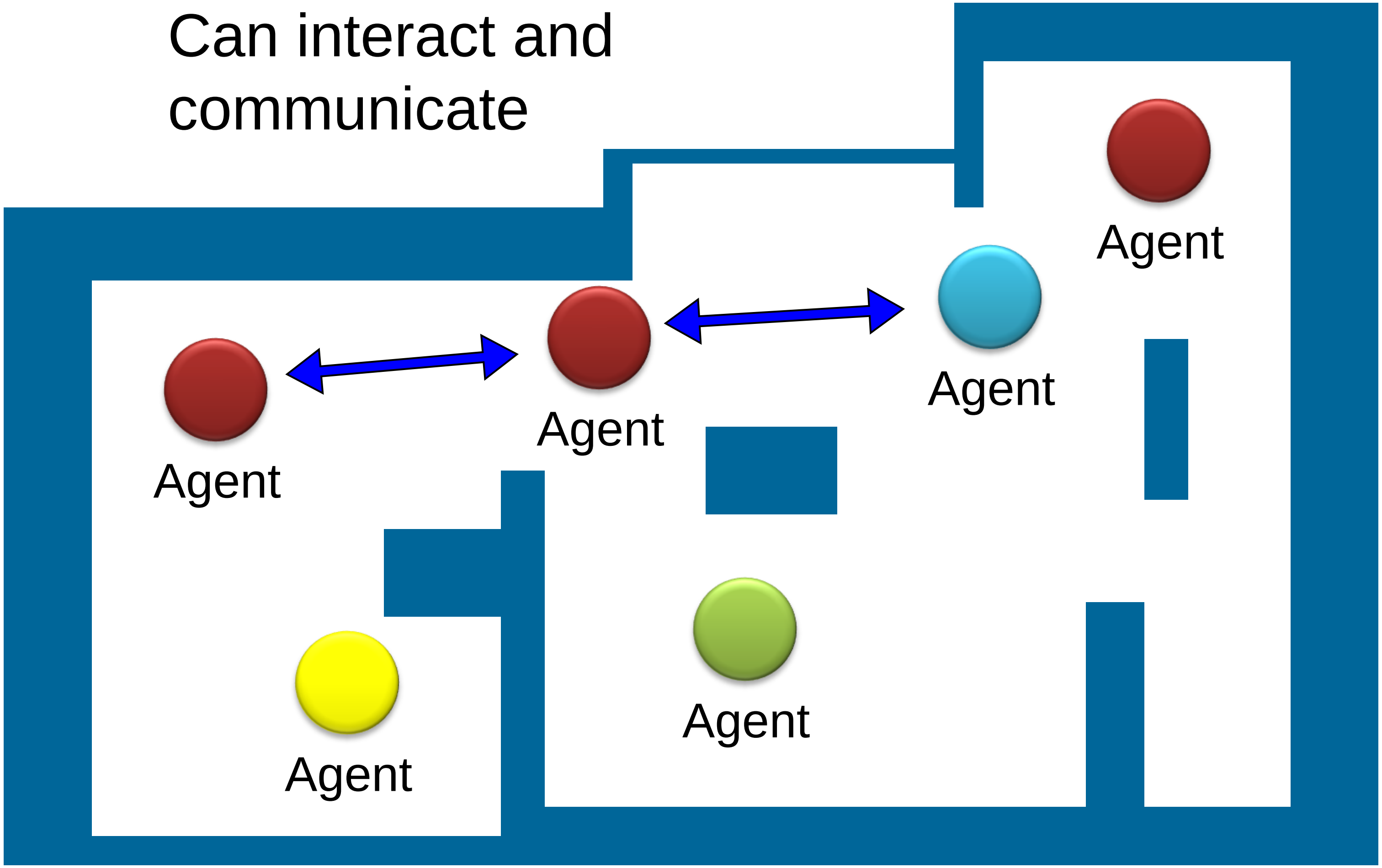
Cohabitate an environment with other agents



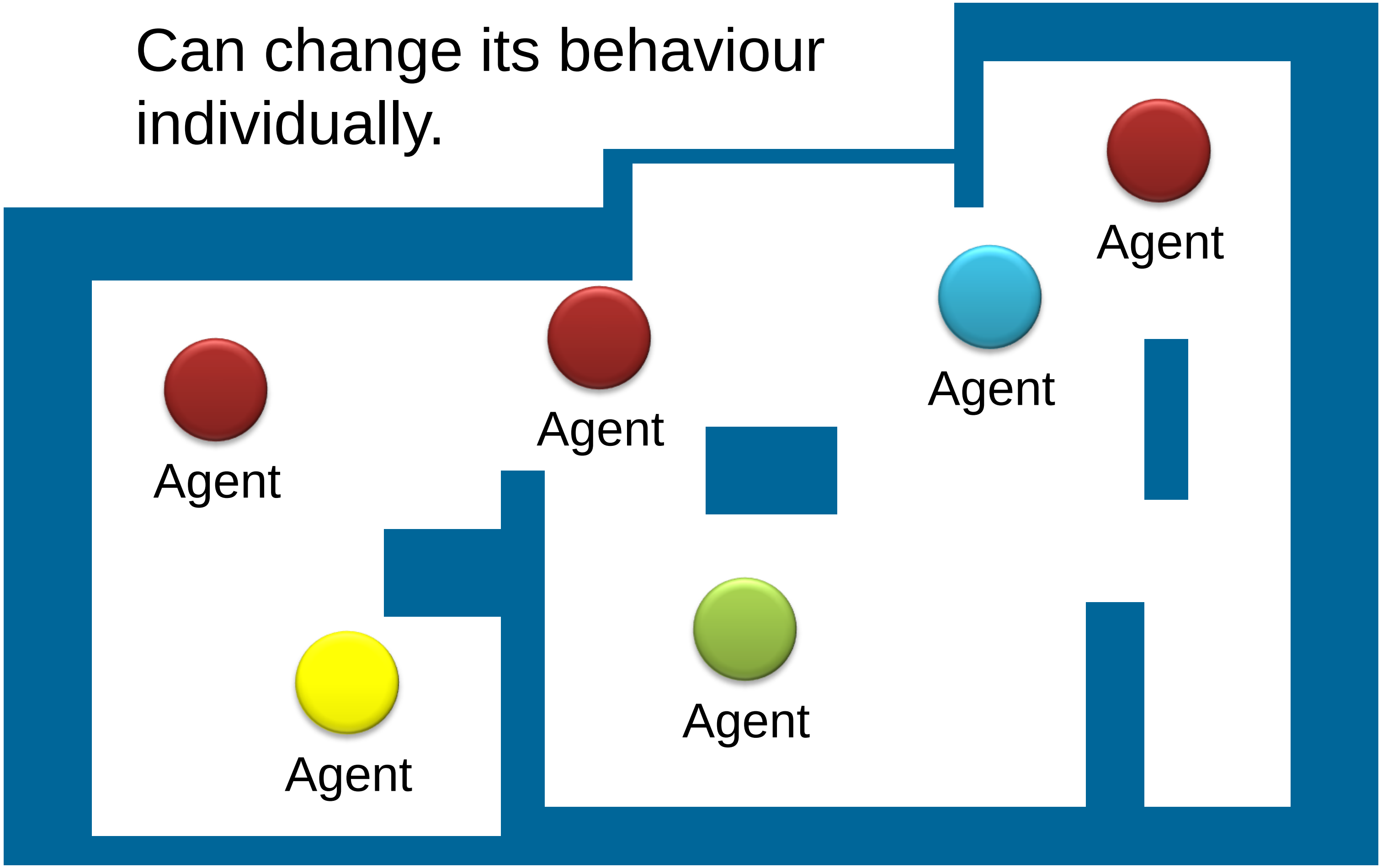

Uniquely Identifiable
Agent
Agent
Agent
Agent
Agent
Agent


Uniquely Identifiable
Agent
Agent
Agent
Agent
Agent
Agent

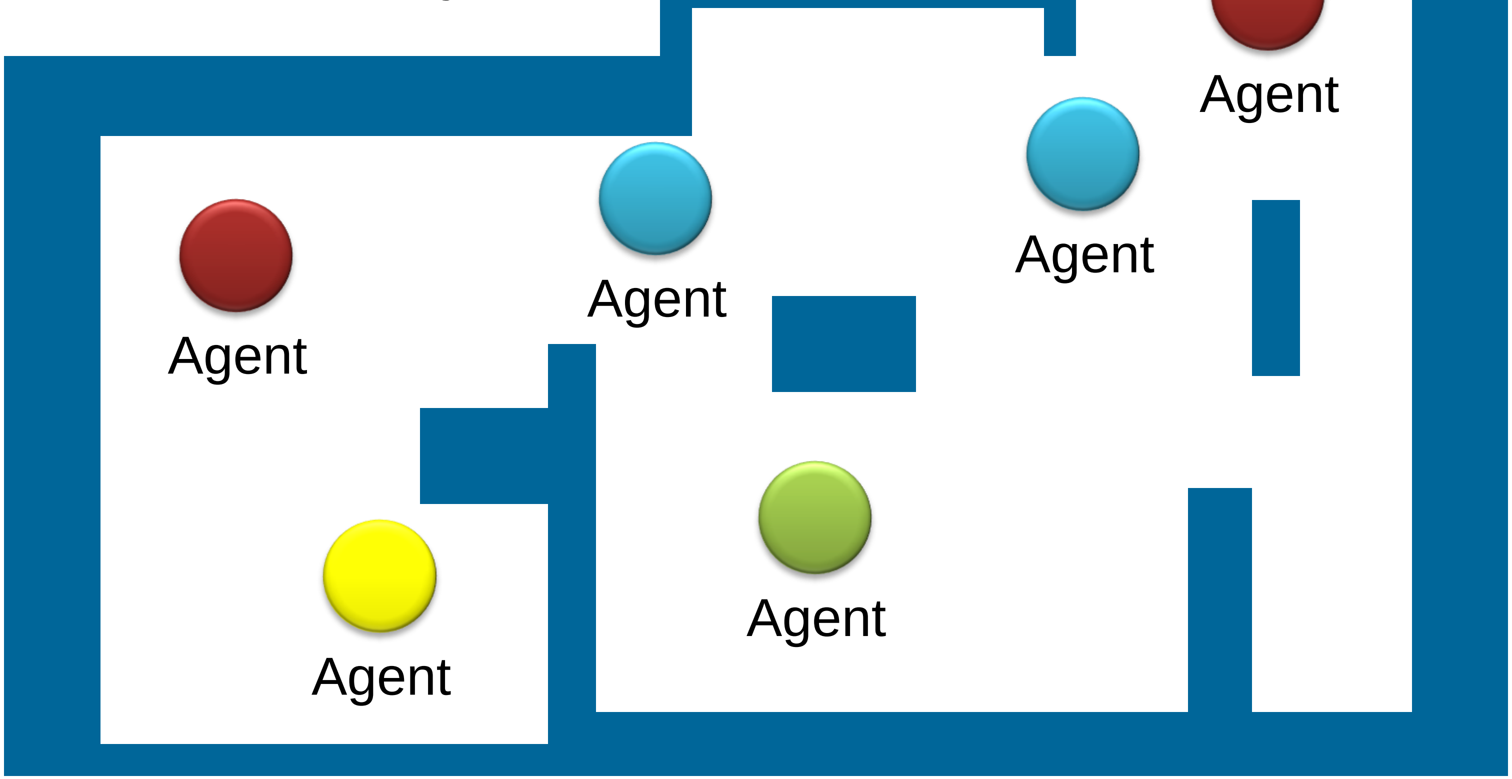
Can interact and communicate



Can change its behaviour individually.



Can change its behaviour individually.



Short Summary

Mathematical Modelling and Simulation

- 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

Mathematical Modelling and Simulation

a. **Representation of „emergent phenomena“**

b. **Flexibility**
   (Bonabeau, 2002)

c. **Natural description of the system**

Properties of Agent-Based Models

Mathematical Modelling and Simulation

a. **Representation of „emergent phenomena“**

b. **Flexibility**
(Bonabeau, 2002)

c. **Natural description of the system**

Representation of „Emergent Phenomena“

Mathematical Modelling and Simulation

Simple rules for individual agents

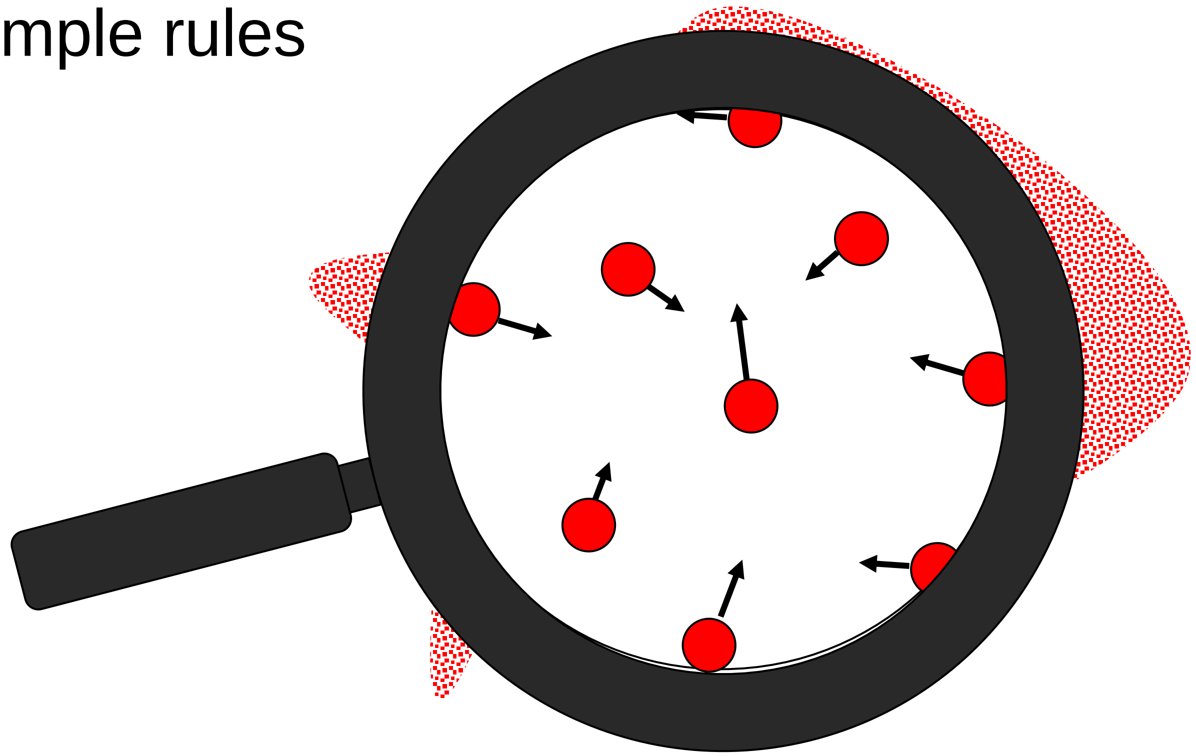
⬇

Complex dynamics of the whole system

group dynamics / swarm intelligence

Representation of „Emergent Phenomena“

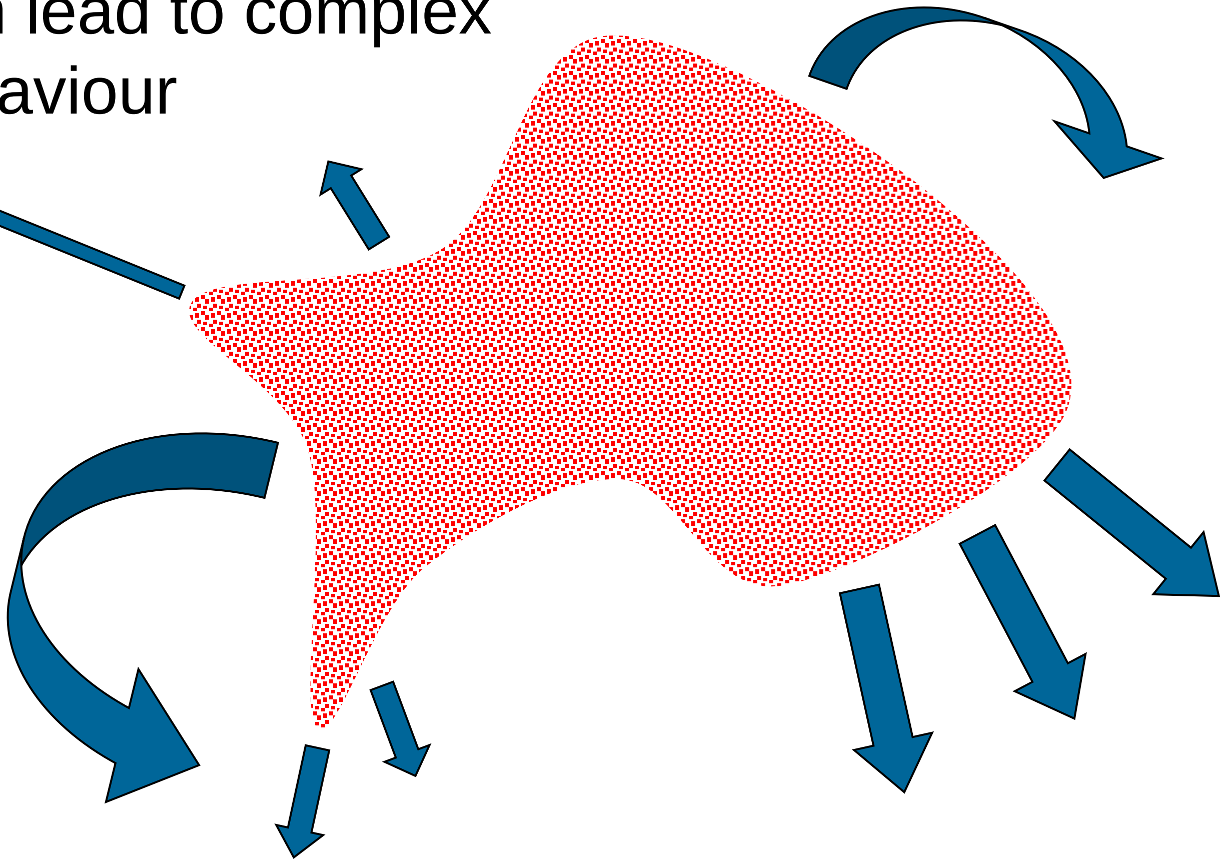
Simple rules



Representation of „Emergent Phenomena“

Mathematical Modelling and Simulation

Can lead to complex behaviour

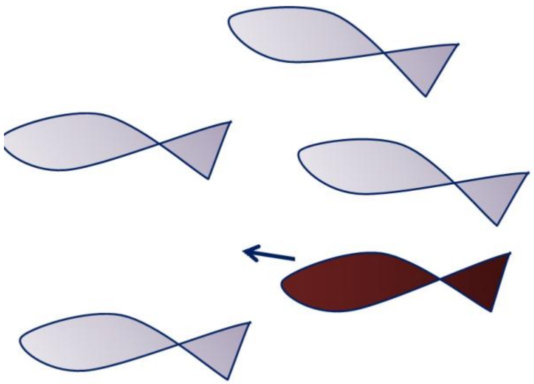


Example: Fish or bird flocks

Mathematical Modelling and Simulation

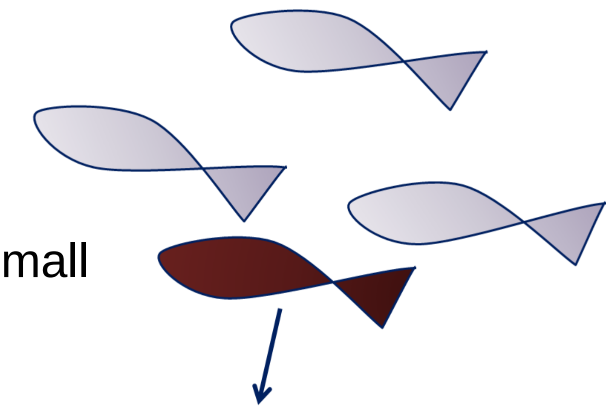
https://www.youtube.com/watch?v=QOGCSBh3kmM

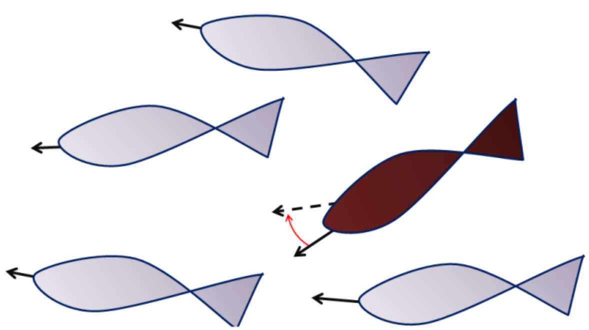
Boids Flock Model



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

Wilensky, U. (1998). NetLogo Flocking model. http://ccl.northwestern.edu/netlogo/models/Flocking. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.

Properties of Agent-Based Models

Mathematical Modelling and Simulation

a. **Representation of „emergent phenomena“**

b. **Flexibility**
(Bonabeau, 2002)

c. **Natural description of the system**

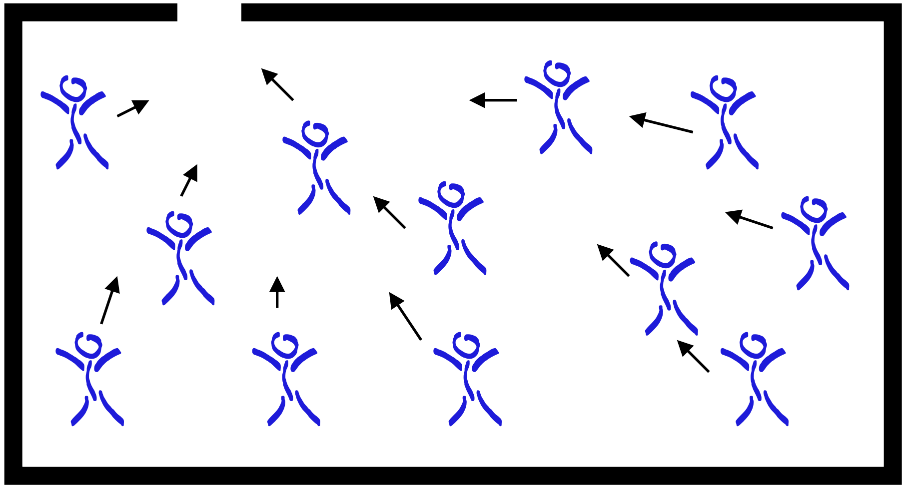
Flexibility

Mathematical Modelling and Simulation

- 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

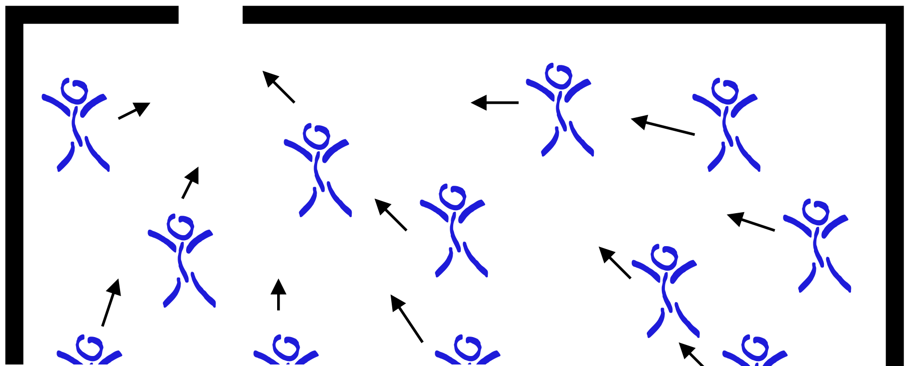
Macroscopic approach

Example: Emergency exit strategy

Mathematical Modelling and Simulation

Agent-Based Model

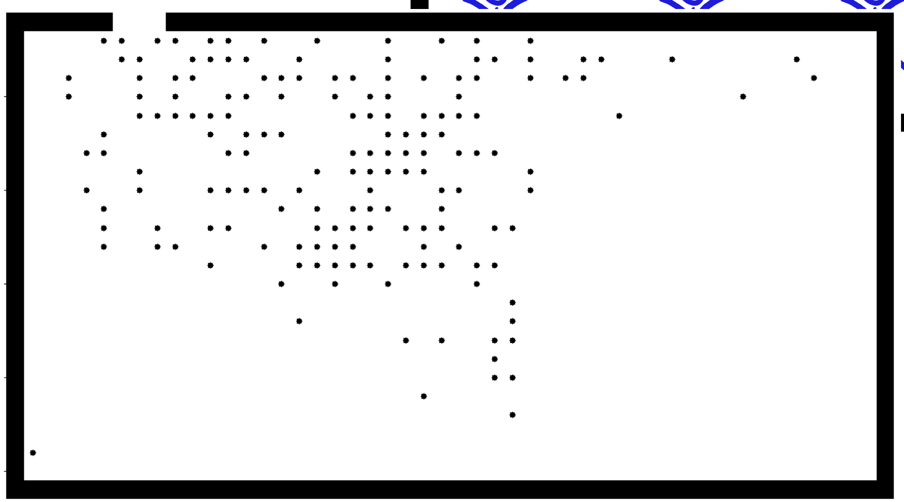
Easy

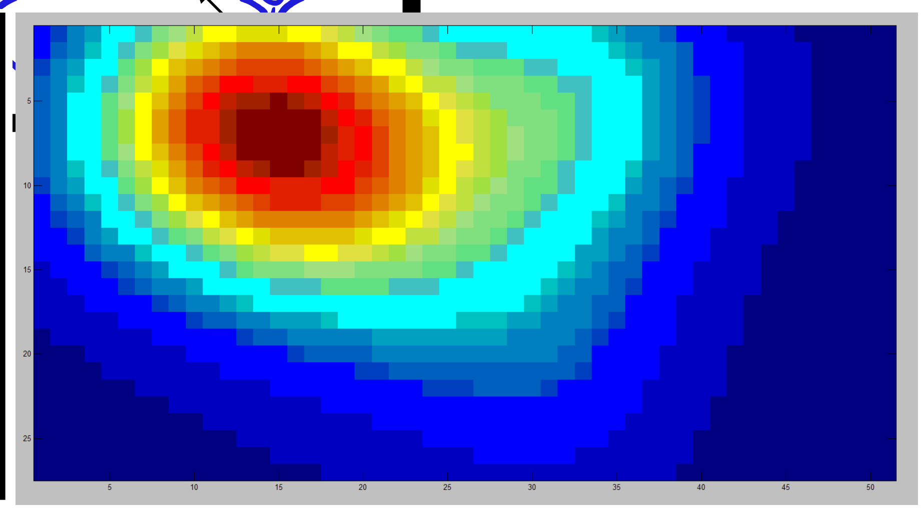


Macroscopic approach

Easy

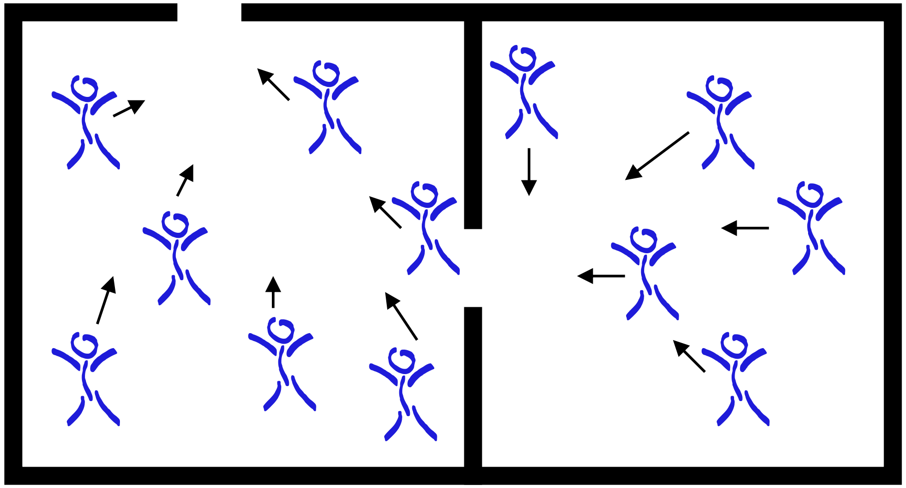
(Navier Stokes PDE Based Model)





Example: Emergency exit strategy

Mathematical Modelling and Simulation



Agent-Based Model

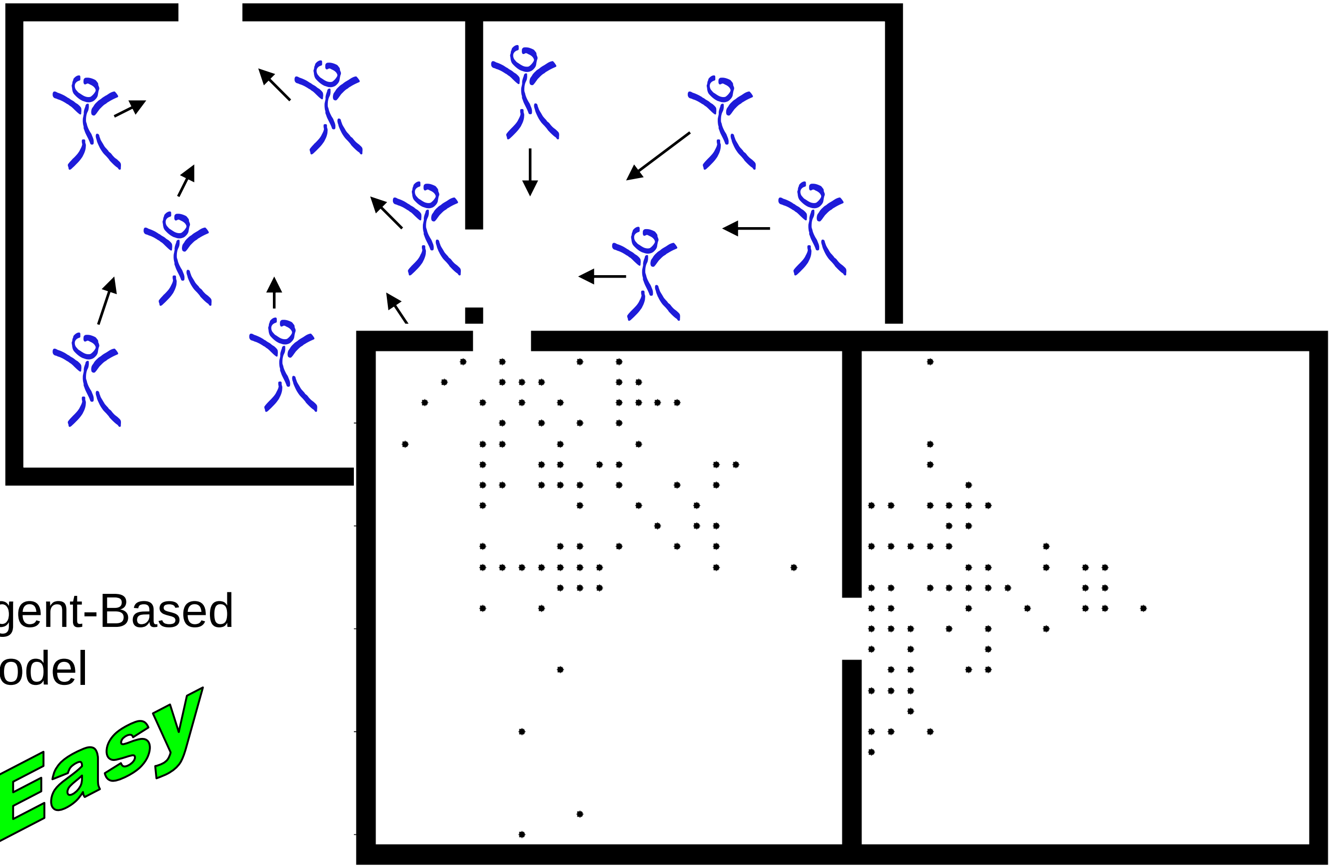
Easy

Macroscopic approach

Tricky

Example: Emergency exit strategy

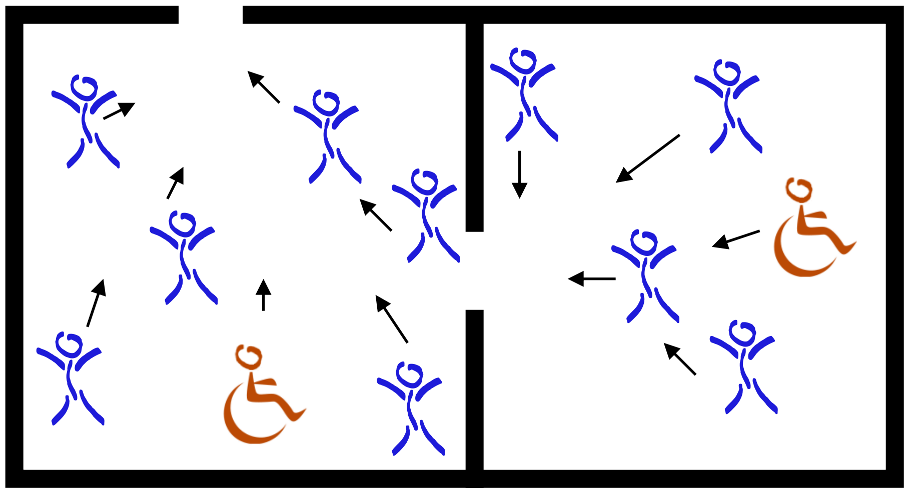
Mathematical Modelling and Simulation



Agent-Based Model

Easy

Example: Emergency exit strategy



Agent-Based Model

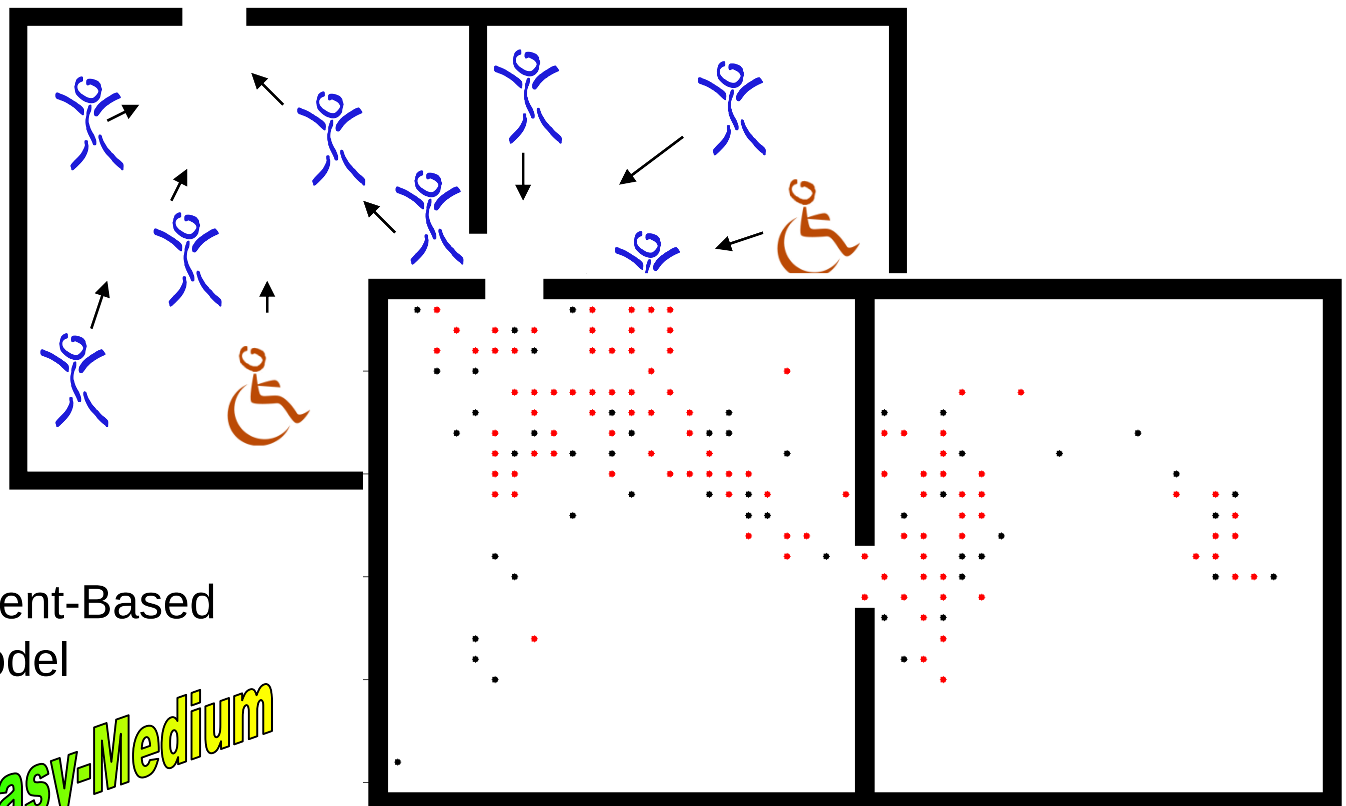
Easy-Medium

Macroscopic approach

Almost Impossible

Example: Emergency exit strategy

Mathematical Modelling and Simulation



Agent-Based Model

Easy-Medium

Properties of Agent-Based Models

Mathematical Modelling and Simulation

a. **Representation of „emergent phenomena"**

b. **Flexibility**
   (Bonabeau, 2002)

c. **Natural description of the system**

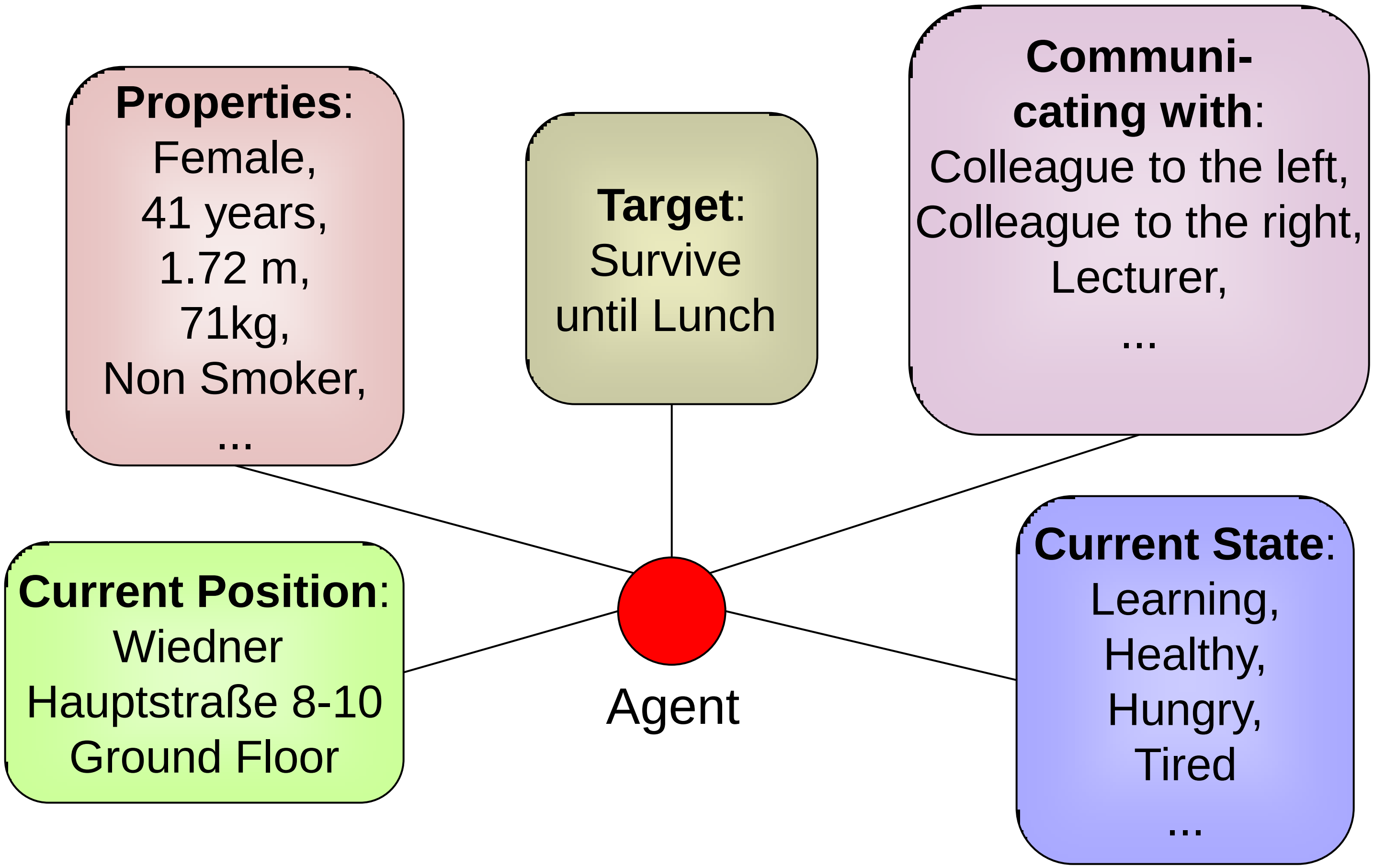
Natural description of the System

Mathematical Modelling and Simulation

- 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

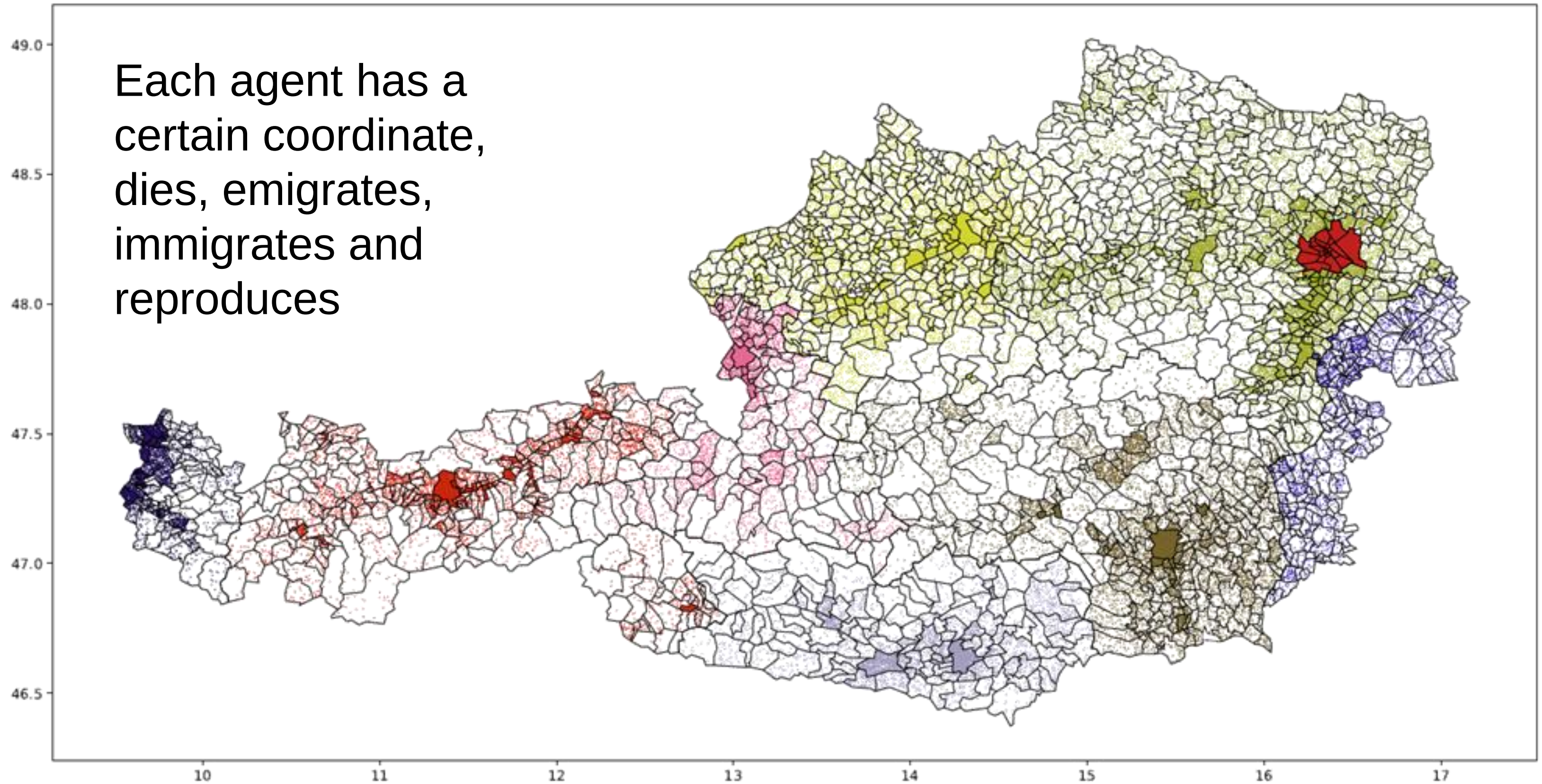
Mathematical Modelling and Simulation



Example: GEPOC (Generic Population Concept)

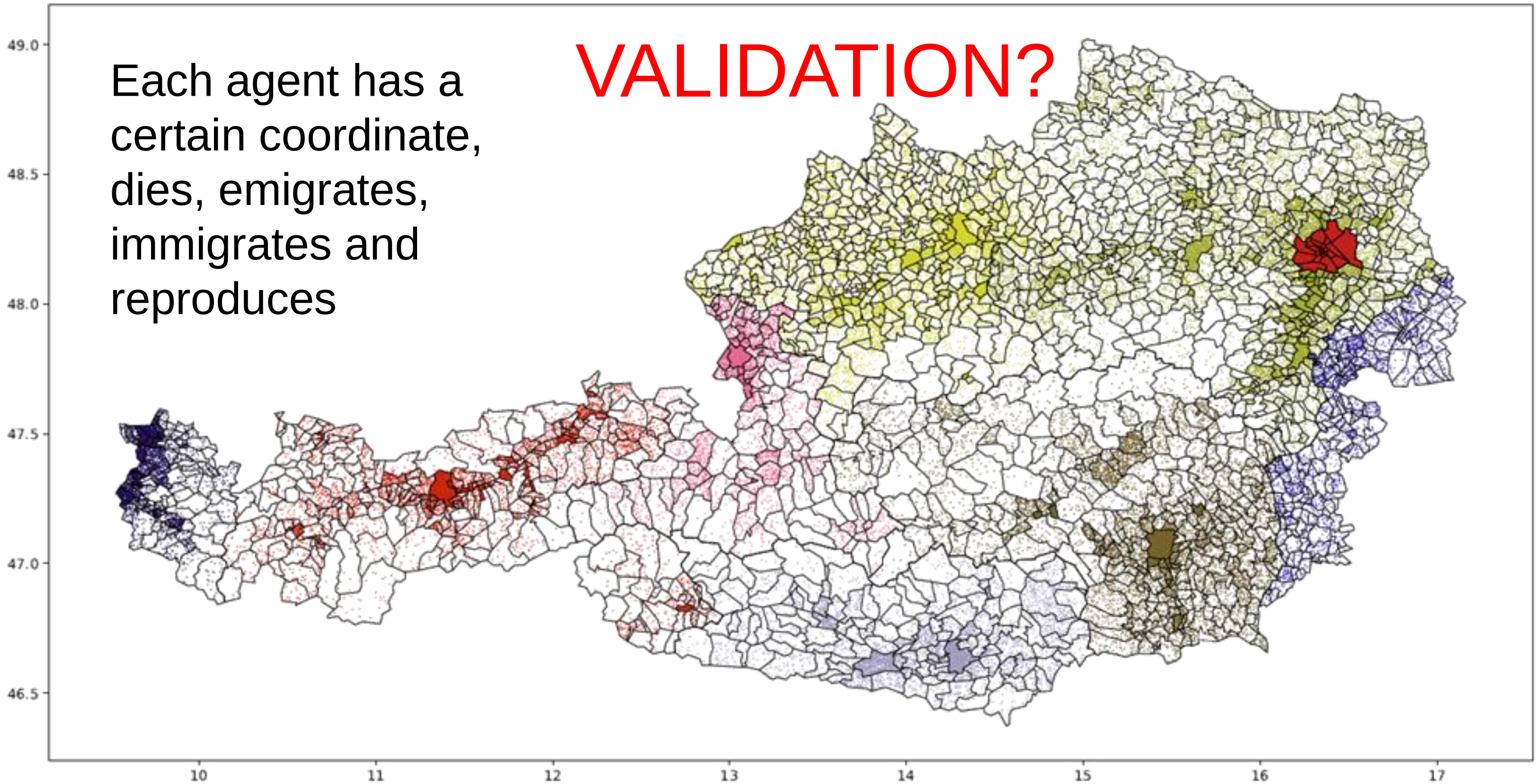
TU WIEN Mathematical Modelling and Simulation

- Population model of Austria
- Simulation of Austria's population from 1999 to make prognosis until 2050

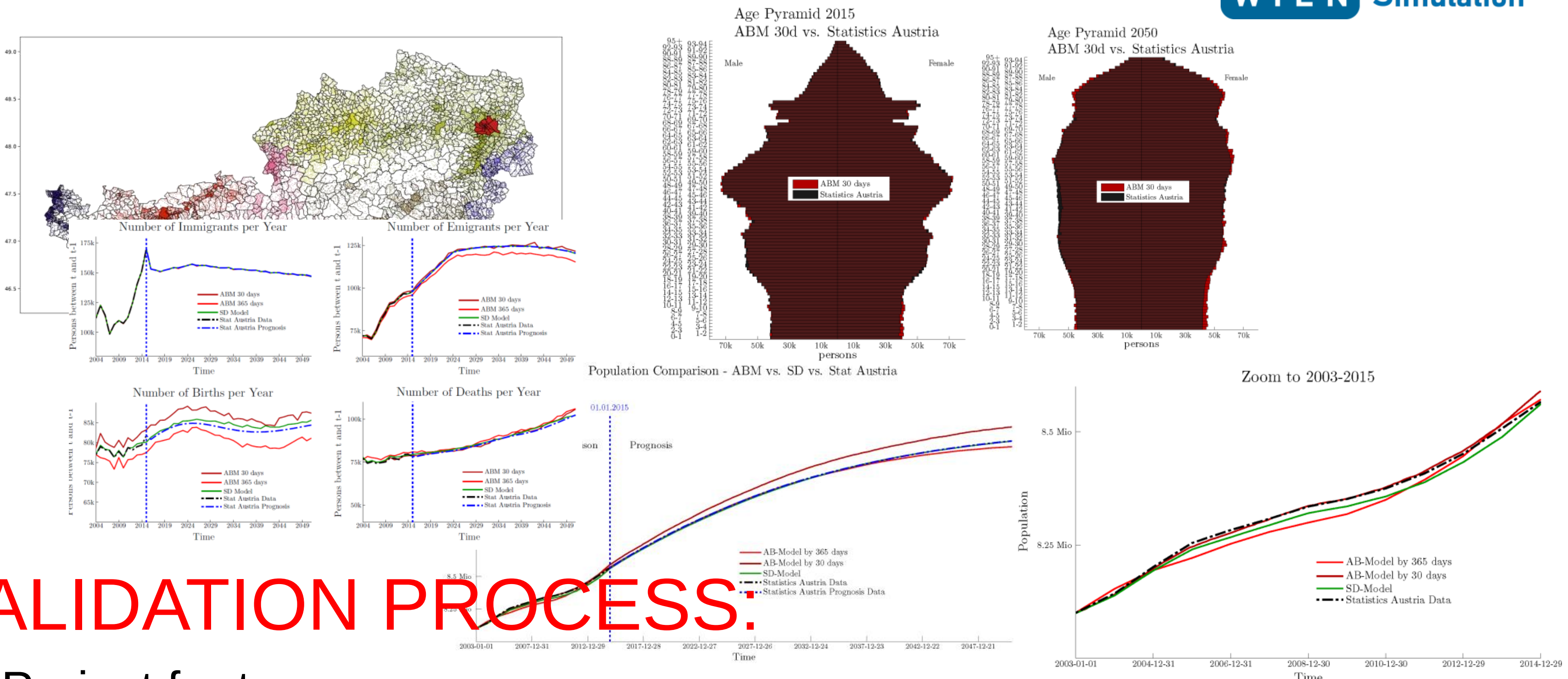


Example: GEPOC (Generic Population Concept)

- Population model of Austria
- Simulation of Austria's population from 1999 to make prognosis until 2050



Example: GEPOC (Generic Population Concept)



VALIDATION PROCESS:

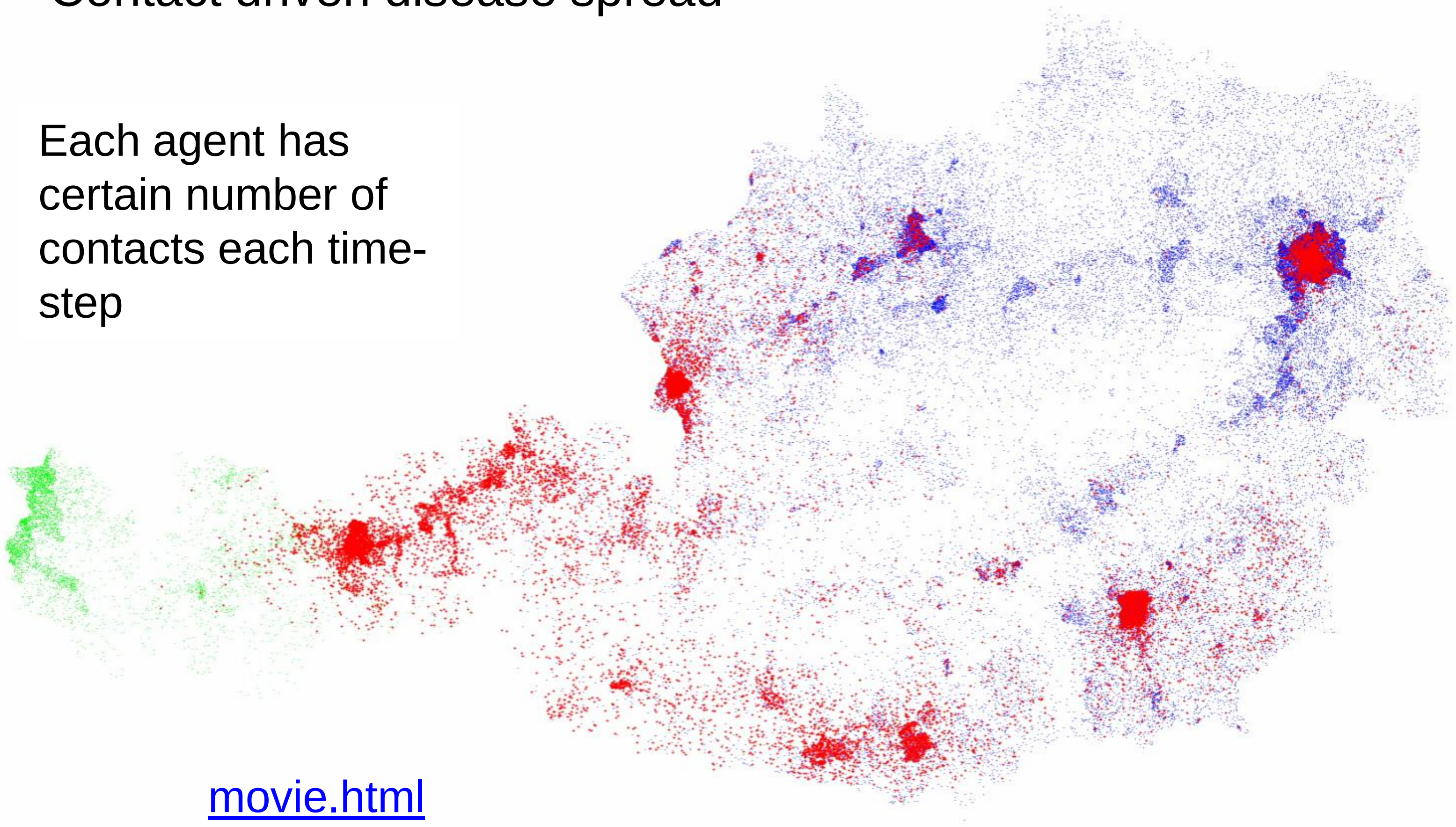
- Project for two years.
- Parametrisation and Validation data for time <2016 from Statistics Austria
- Parametrisation and Validation for time >=2016 matched with Statistics Austria Prognosis tool

Example: GEPOC Flu

Mathematical Modelling and Simulation

- Simulation of 2014 Flu
- Contact driven disease spread

Each agent has certain number of contacts each time-step



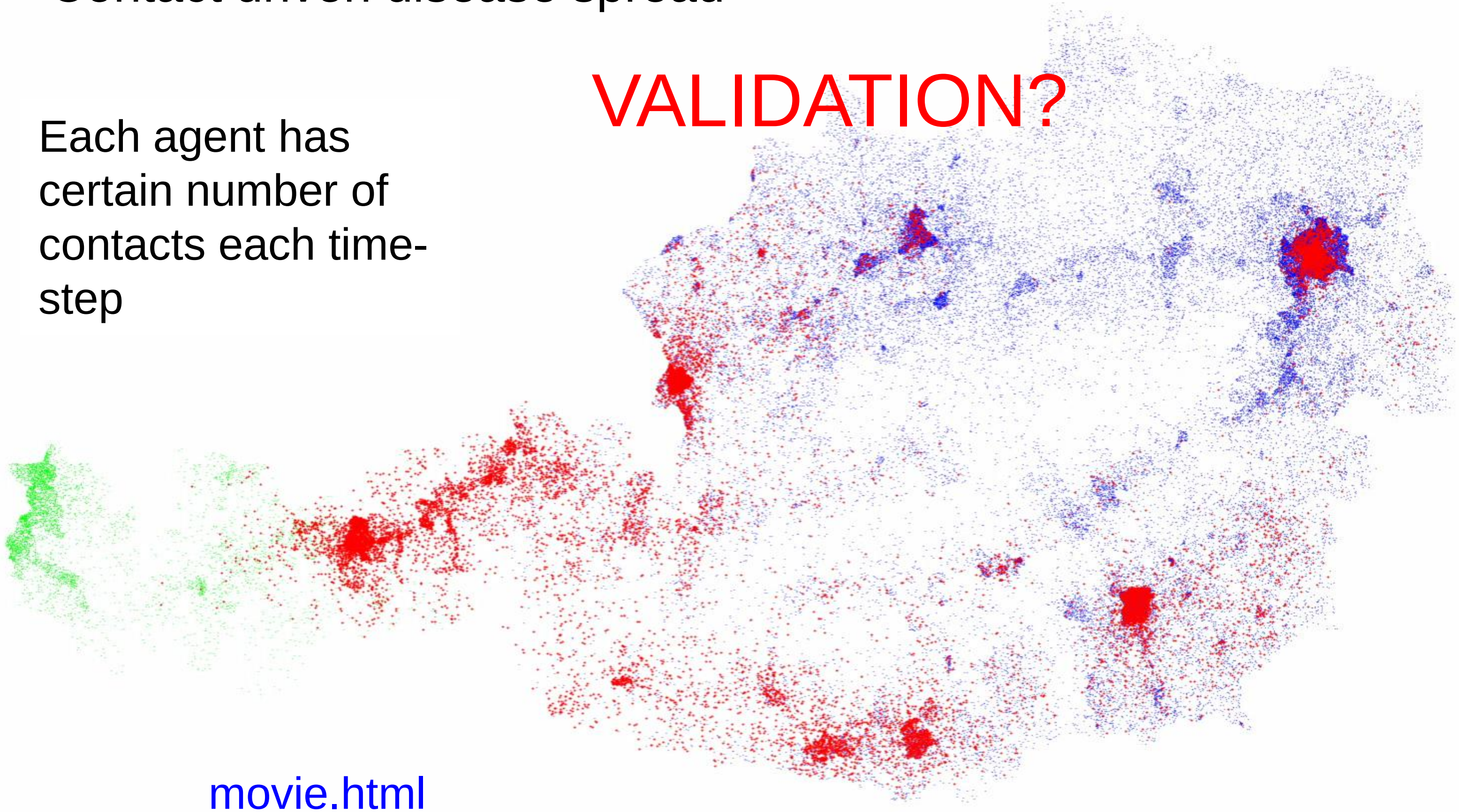
movie.html

Example: GEPOC Flu

- Simulation of 2014 Flu
- Contact driven disease spread

VALIDATION?

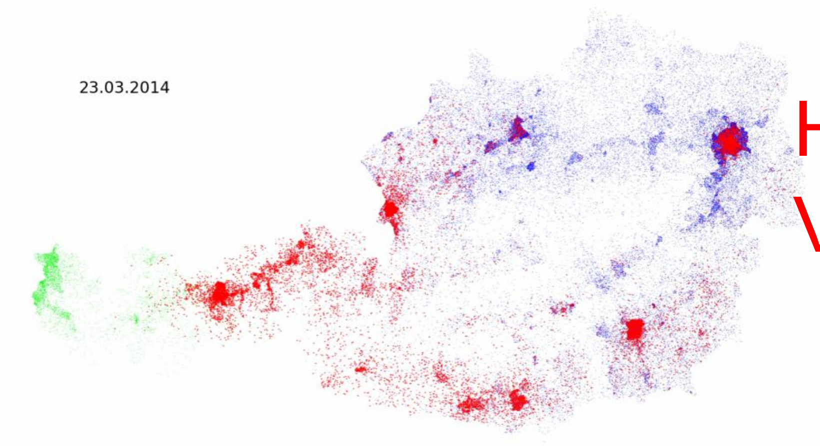
Each agent has certain number of contacts each time-step



movie.html

Example: GEPOC Flu

Mathematical Modelling and Simulation



HOW ABOUT VALIDATION?

THIS model is absolute rubbish and has hardly anything to do with reality!

Beware of wrong ideas!

- Natural desripion of the system makes the model easier to communicate.
- Therefore it becomes more credible than more abstract approaches

BUT

CREDIBLE ≠ VALID

PICTURESQUE≠ VALID

Interpretation of Agent-Based Model Results

Mathematical Modelling and Simulation

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

Interpretation of Agent-Based Model Results

Basically two classes of agent-based models can be observed

ABMs for **qualitative** investigation

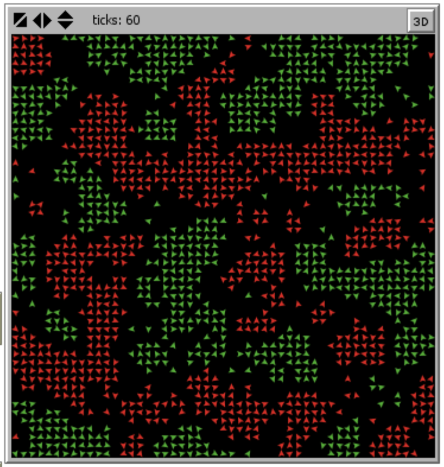
- (On purpose) very abstract
- Usually very complex model behaviour
- Hardly any parameters identified with real data

ABMs for **quantitative** investigation

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

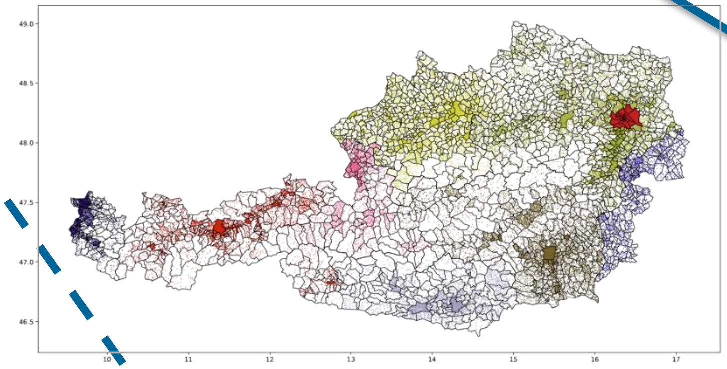
Interpretation of Agent-Based Model Results : Examples

ABMs for **qualitative** investigation



Schelling‘s Segregation Model

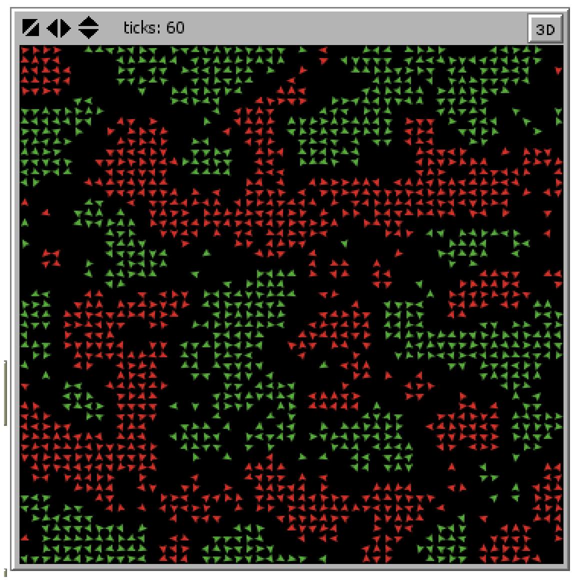
ABMs for **quantitative** investigation



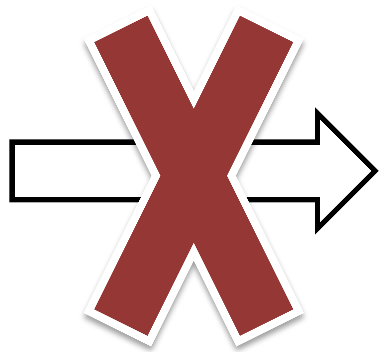
GEPOC

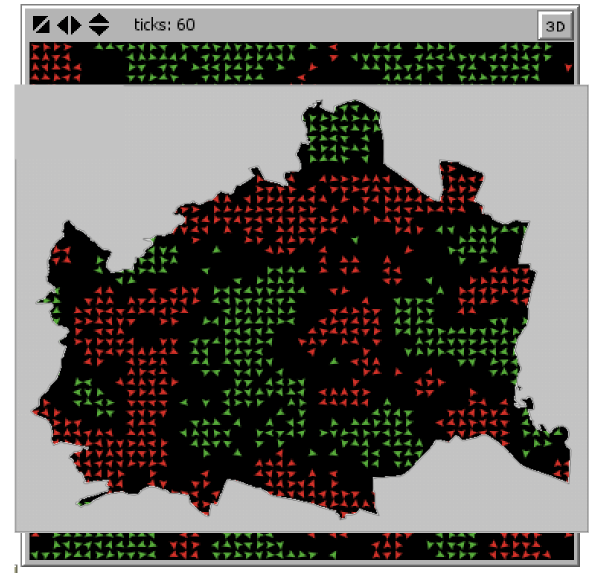
Interpretation of Agent-Based Model Results : Examples

Mathematical Modelling and Simulation



WRONG INTERPRETATION

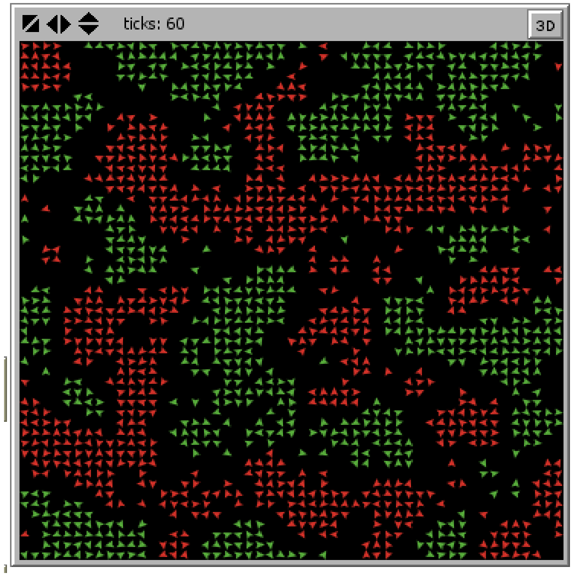




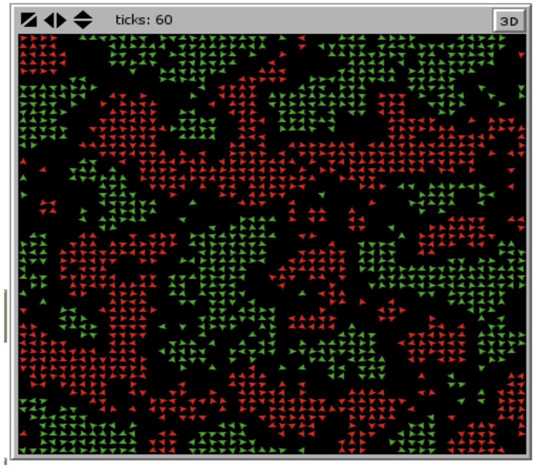
„Schelling's model predicts: In a few years only immigrants in Wien Hietzing!"

Interpretation of Agent-Based Model Results : Examples

Mathematical Modelling and Simulation



CORRECT INTERPRETATION

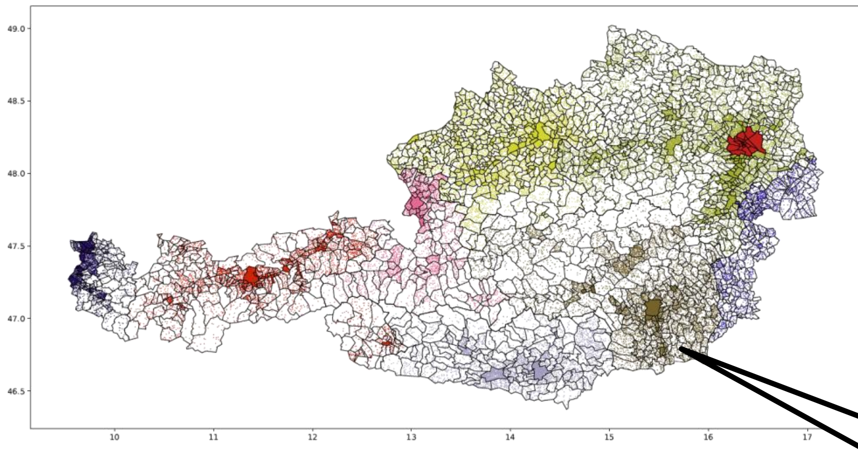


„If we do not take care on our migration policy human homophobia might lead to spatially visible ghettoism as seen above in Austria as well!“

Interpretation of Agent-Based Model Results : Examples

Mathematical Modelling and Simulation

WRONG INTERPRETATION

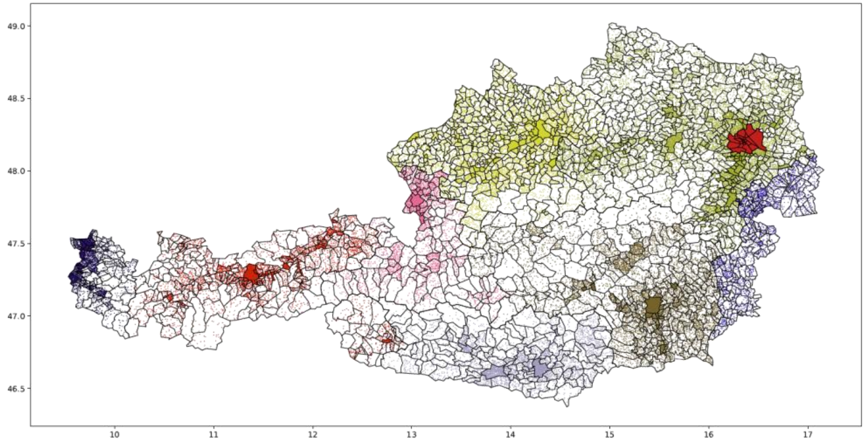


„GEPOC predicts: In two years there will be a 50 year old immigrant in Leibnitz“

Hi guys, i‘m Mike

In general: **Never** 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: 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: 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.

Mathematical Modelling and Simulation

Questions?

Discrete Modelling
Difference Equations

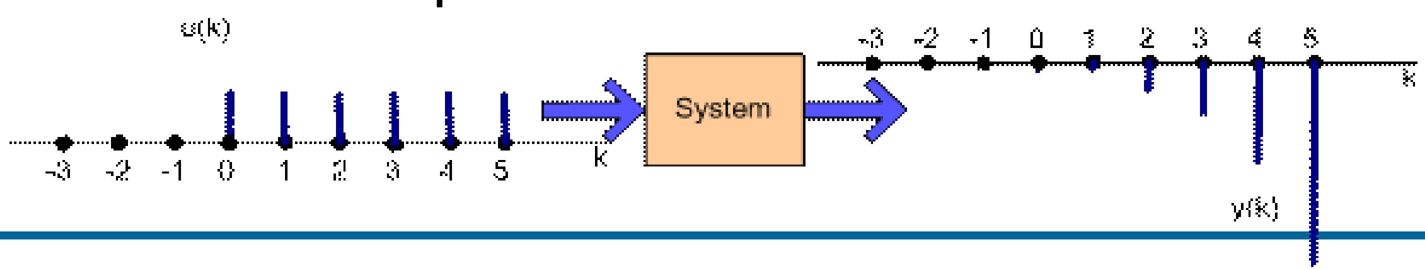
Part 1

Difference Equation

Mathematical Modelling and Simulation

- equations involving differences of inputs and outputs
- three points of views
  - sequence of number
  - discrete dynamical system
  - iterated function

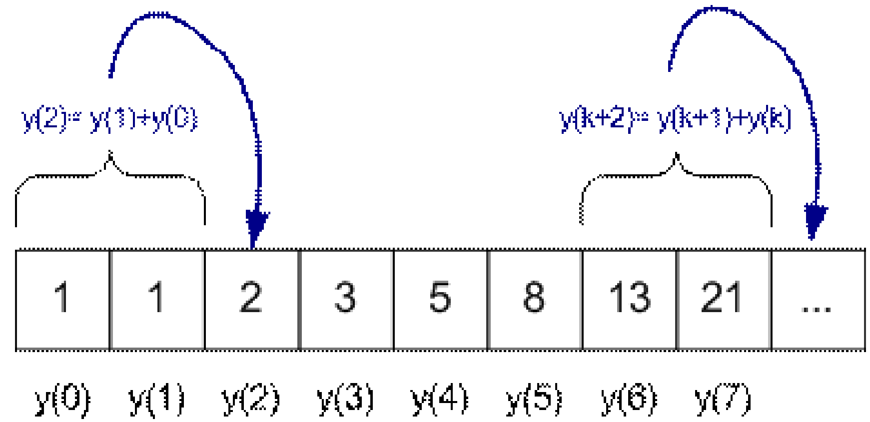
Difference equation - is a sequence of numbers that generated recursively using a rule to relate each number in the (output) sequence to previous (output) numbers and input numbers in the sequence.



Difference Equations

- **Fibonacci Sequence :**

  $\{1,1,2,3,5,8,13,21,34\}$
  $y(k+2) = y(k+1) + y(k)$
  $y(0) = y(1) = 1, k = 0,1, \ldots$

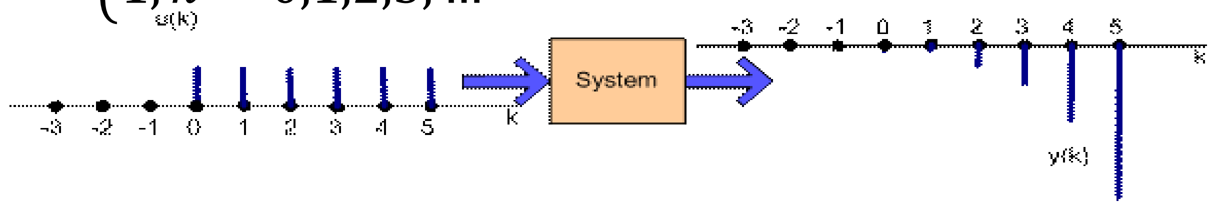


- **Growth model**

- **Dynamical System with unit step input**

$$y(k) = 2y(k-1) + \frac{3}{2}u(k)$$

$$u(k) = \begin{cases} 0, k = -1, -2, -3, \ldots \\ 1, k = 0,1,2,3, \ldots \end{cases} \quad \Rightarrow \quad y(k) = \frac{3}{2}(1 - 2^{k+1})$$



Difference Equations

Mathematical Modelling and Simulation

- Iterated map $f(k)$
  $y(k+2) = f\big(y(k)\big), y(0) = y_0, k = 0,1,2,3, \dots$

  orbit $\{y_0, f(y_0), f\big(f(y_0)\big), f\Big(f\big(f(y_0)\big)\Big), \dots\}$
  dependent on y_0

- Example: $y(k+1) = f\big(y(k)\big) := y(k)^2, y(0) = y_0, k = 0,1,2,3 \dots$

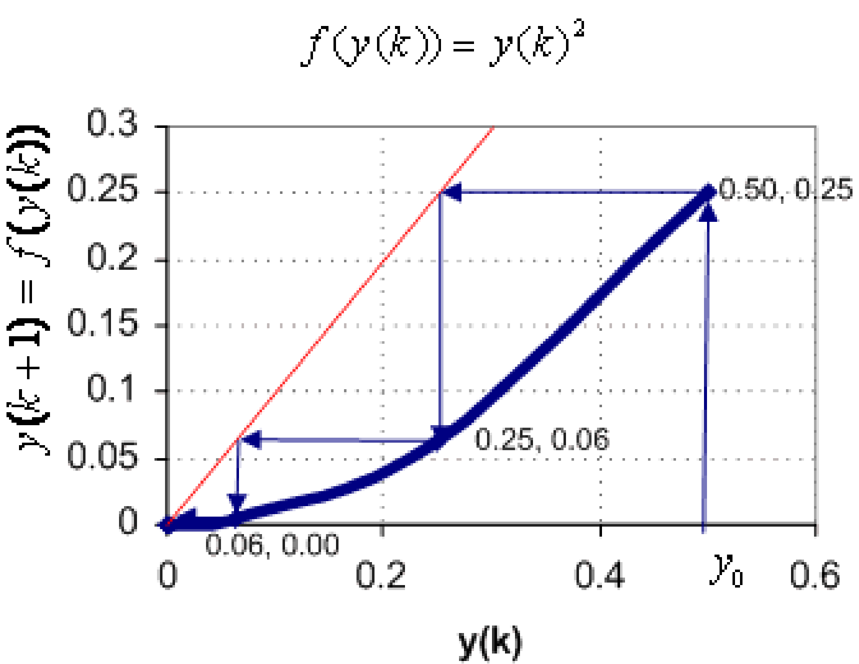
$$y(0) = 1, \Rightarrow orbit\ \{1,1,1,1, \dots\}$$
$$y(0) = -1 \Rightarrow orbit\ \{-1,1,1,1, \dots\}$$
$$y(0) = 2 \Rightarrow \ orbit\ \{2,4,16,256,65536, \dots\}$$
$$y(0) = \frac{1}{2} \Rightarrow \ orbit\ \{0.5,0.25,0.0625, 0.00390625, \dots\}$$

Difference Equations

TU WIEN Mathematical Modelling and Simulation

- Example $y(k+1) = f\big(y(k)\big) = y(k)^2, y(0) = y_0, k = 0,1,2,3$

$$y(0) = \frac{1}{2} \Rightarrow \quad orbit\ \{0.5, 0.25, 0.0625, 0.00390625, \dots\}$$



Cobweb Function:

$$(y(0),0) \rightarrow \big(y(0), y(1)\big) \rightarrow$$
$$\rightarrow (y(1), y(1)) \rightarrow \big(y(1), y(2)\big) \rightarrow$$
$$\rightarrow (y(2), y(2)) \rightarrow \big(y(2), y(3)\big) \rightarrow$$
$$\rightarrow (y(3), y(3)) \rightarrow \big(y(3), y(4)\big) \rightarrow$$
$$\dots$$

„oscillates" between $y = f(x)$ and $y = x$

Difference Equations

Mathematical Modelling and Simulation

- **Equlibria – Fixed Points**
  $y(k+2) = f\big(y(k)\big), y(0) = y_0, k = 0,1,2,3, \ldots$
  $Equilibrium\ y^*: \quad y^* = f(y^*) \Leftrightarrow y(k+1) = f\big(y(k)\big) = y(k)$
- **Attractive/stable:** $y_0, y_1, y_2, y_3, \ldots.\ converge\ to\ y^*$
- **Repelling/unstable:** $y_0, y_1, y_2, y_3, \ldots.\ diverge\ from\ \ y^*$

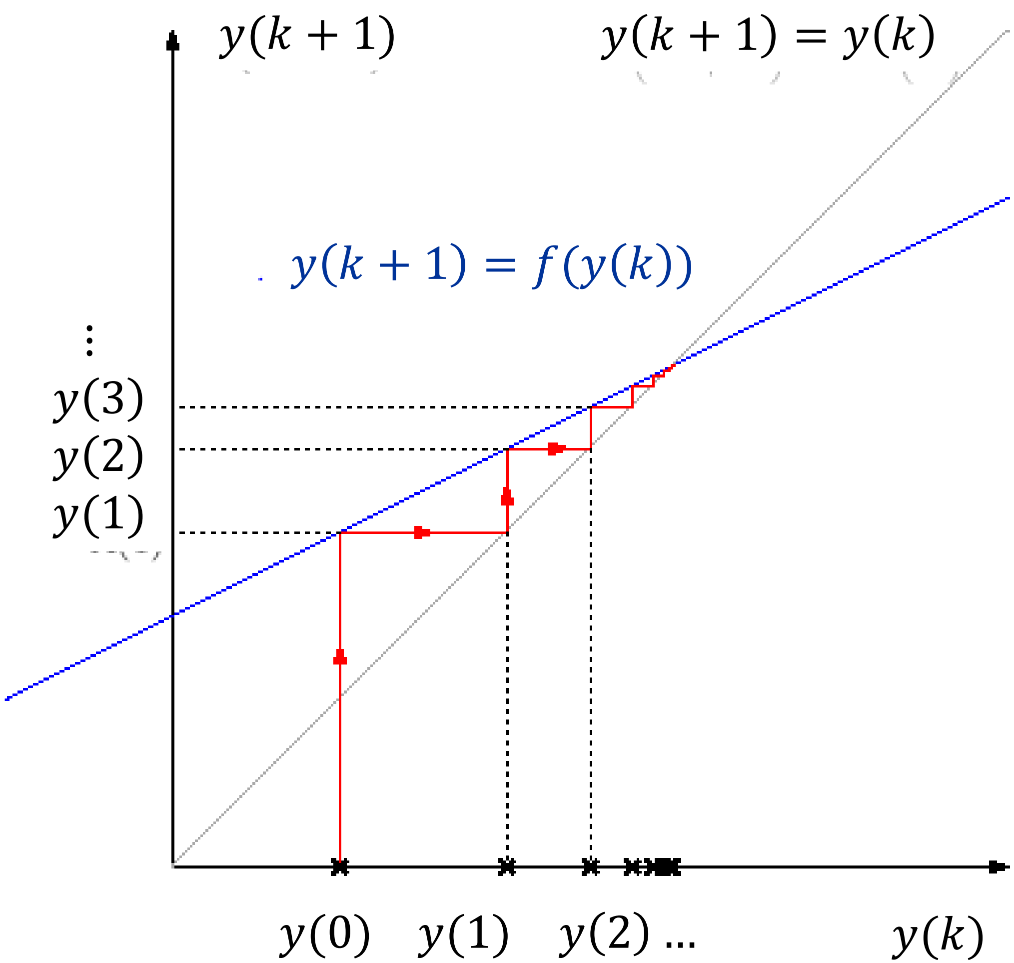
- **Graphic Test for stability / instability:**

  Cobweb-function stable/attractive:
  $(y(0), 0) \to \big(y(0), y(1)\big) \to (y(1), y(1)) \to \big(y(1), y(2)\big)$
  $\to (y(2), y(2)) \to \big(y(2), y(3)\big) \to \cdots \to (y*, y*)$

  Cobweb–function stable/attractive:
  $(y(0), 0) \to \big(y(0), y(1)\big) \to (y(1), y(1)) \to \big(y(1), y(2)\big)$
  $\to (y(2), y(2)) \to \big(y(2), y(3)\big) \to \cdots diverge$

Difference Equations

Mathematical Modelling and Simulation



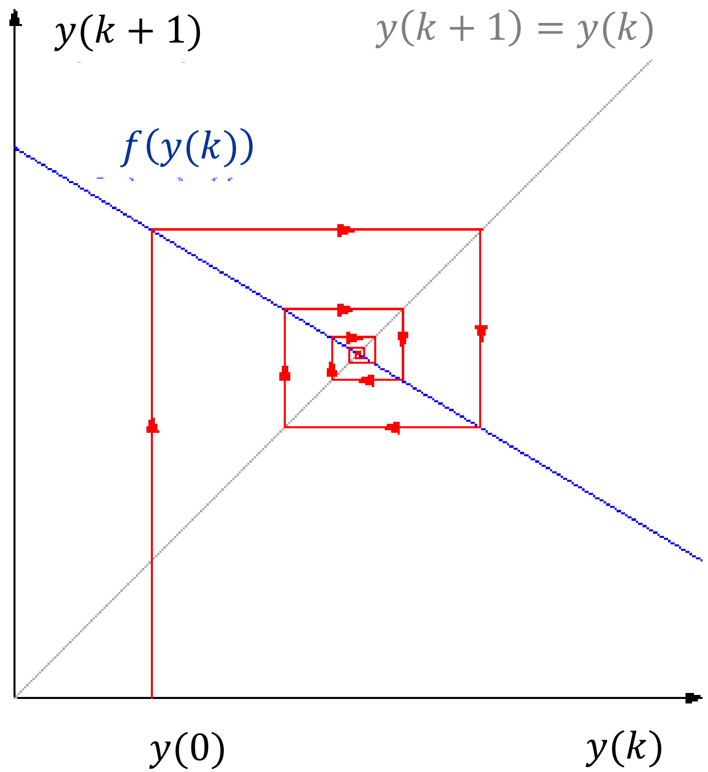
$$y(k+1) = \frac{4}{9}y(k) + \frac{7}{2}$$

Cobweb Diagram

- Graphical technique to investigate iterated functions
- Iteration is performed graphically
- Consists of
  - Iterated Function $f(y)$
  - 1.Mediane $y(k+1) = y(k)$
  - Cobweb path

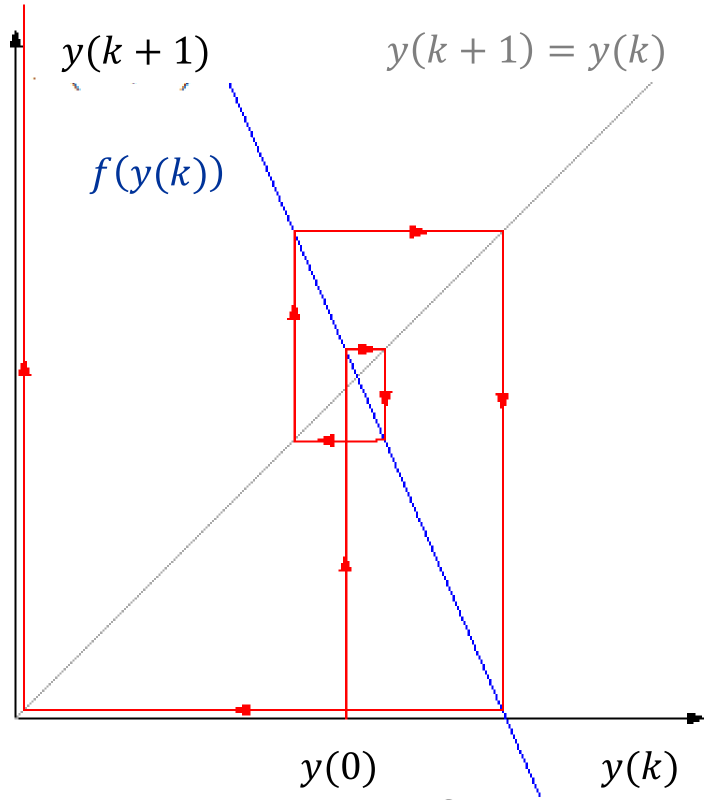
Cobweb Functions

$$y(k+1) = -0.6y(k) + 8$$



Inward spirals lead to attracting fixed points

$$y(k+1) = -3.5y(k) + 17.5$$



Outward spirals from repelling fixed points

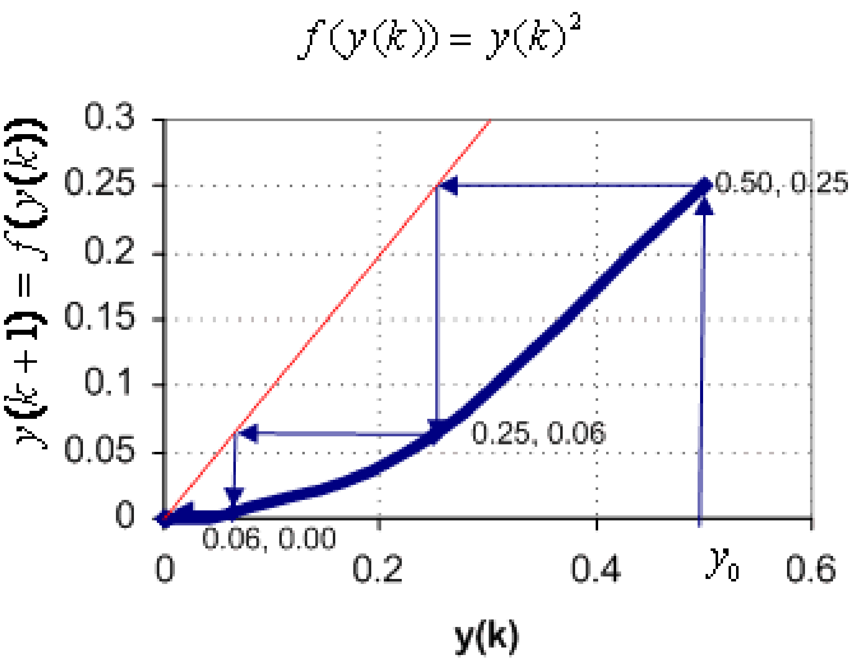
Cobweb Functions

TU WIEN Mathematical Modelling and Simulation

- Example

$$y(k+1) = f\big(y(k)\big) := y(k)^2, y(0) = y_0, k = 0,1,2,3 \ldots$$

$$\Rightarrow Equilibria \;\; y^* = f(y^*) = {y^*}^2 \Rightarrow y^* \;\in \{0,1\}$$



Cobweb Function:

$$(y(0), 0) \to \big(y(0), y(1)\big) \to$$
$$\to (y(1), y(1)) \to \big(y(1), y(2)\big) \to$$
$$\to (y(2), y(2)) \to \big(y(2), y(3)\big) \to$$
$$\to (y(3), y(3)) \to \big(y(3), y(4)\big) \to$$
$$\ldots \to (0,0)$$

attracts $y^* = 0$

Linear Affine Difference Equations

$$y(k+1) = f\big(y(k)\big) = ay(k) + b, y(0) = y_0, k = 0,1,2, \ldots$$

- Examples in Finance
  - Actual balance y(n)
    - after n compounding periods
    - with annual interest I
    - compounded m times a year
    - and constant amount b added at the end of every compounding period:

$$y(n+1) = \left(1 + \frac{I}{m}\right) y(n) + b$$

Linear Affine Difference Equations

Mathematical Modelling and Simulation

$$y(k+1) = f\big(y(k)\big) = ay(k) + b, y(0) = y_0, k = 0,1,2,\dots$$

- **Solution**

  $y(1) = ay(0) + b = ay_0 + b$

  $y(2) = ay(1) + b = a(ay_0 + b) + b = a^2y_0 + ab + b$

  $y(3) = ay(2) + b = a(a^2y_0 + ab + b) + b$

  $= a^3y_0 + (a^2 + a + 1)b$

  ...

  $$y(k) = a^ky_0 + \left(1 + a + a^2 + \dots + a^{k-1}\right)b = a^ky_0 + b\sum_{i=0}^{k-1} a^i$$

Linear Affine Difference Equations

Mathematical Modelling and Simulation

$$y(k+1) = f\big(y(k)\big) = ay(k) + b, y(0) = y_0, k = 0,1,2, \ldots$$

- **Solution**

$$y(k) = a^k y_0 + \left(1 + a + a^2 + \cdots + a^{k-1}\right)b = a^k y_0 + b\sum_{i=0}^{k-1} a^i$$

$\sum_{i=0}^{k-1} a^i$ geometric series for $a \neq 1$

$$\rightarrow \sum_{i=0}^{k-1} a^i = \frac{1-a^k}{1-a}$$

and for $a = 1 \rightarrow \sum_{i=0}^{k-1} a^i = \sum_{i=0}^{k-1} 1 = k$

- Hence

$$y(k) = \begin{cases} a^k y_0 + b\dfrac{1-a^k}{1-a}, & a \neq 1 \\ y_0 + kb, & a = 1 \end{cases}$$

Linear Affine Difference Equations

TU WIEN Mathematical Modelling and Simulation

$$y(k+1) = f\big(y(k)\big) = ay(k) + b, y(0) = y_0, k = 0,1,2,\ldots$$

- **Solution**

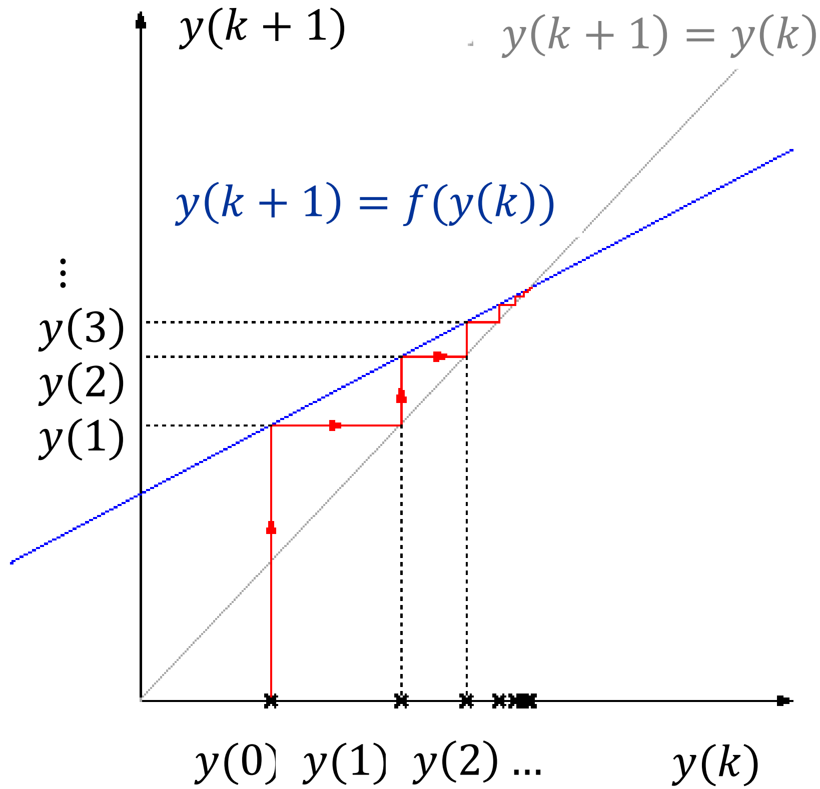
$$y(k) = \begin{cases} a^k y_0 + b\dfrac{1-a^k}{1-a}, a \neq 1 \\ y_0 + bk, \qquad a = 1 \end{cases}$$

Example:

$$y(k+1) = \frac{4}{9}y(k) + \frac{7}{2},$$

$$y(0) = 2.25 = \frac{9}{4}$$

$$y(k) = \frac{9}{4}\frac{4^k}{9^k} + \frac{7}{2}\frac{1-\frac{4^k}{9^k}}{1-\frac{4}{9}} = \frac{(7\cdot 3^{2k-2} - 2^{2n-1})}{10\cdot 3^{2n-4}}$$



Linear Affine Difference Equations

Mathematical Modelling and Simulation

$$y(k+1) = f\big(y(k)\big) = ay(k) + b, y(0) = y_0, k = 0,1,2,\ldots$$

- **Equilibrium / Fixed Point**
- $y^* = f(y^*) \leftrightarrow y^* = ay^* + b$

$$y^* = \frac{b}{1-a}, a \neq 1$$

  - Attractive/stable: $y_0, y_1, y_2, y_3, \ldots.\,converge\ to\ y^*$
  - Repelling/unstable: $y_0, y_1, y_2, y_3, \ldots.\,diverge\ from\ \ y^*$

- Solution with Equilibrium

$$y(k) = a^k y_0 + b\frac{1-a^k}{1-a} = a^k\left(y_0 - \frac{b}{1-a}\right) + \frac{b}{1-a}$$

$$= a^k(y_0 - y^*) + y^*, a \neq 1$$

$$y(k) = y_0 + k\,, a = 1$$

Linear Affine Difference Equations

$$y(k+1) = f\big(y(k)\big) = ay(k) + b, y(0) = y_0, k = 0,1,2, \dots$$

- **Solution**

$$y(k) = a^k y_0 + b\frac{1-a^k}{1-a} =$$

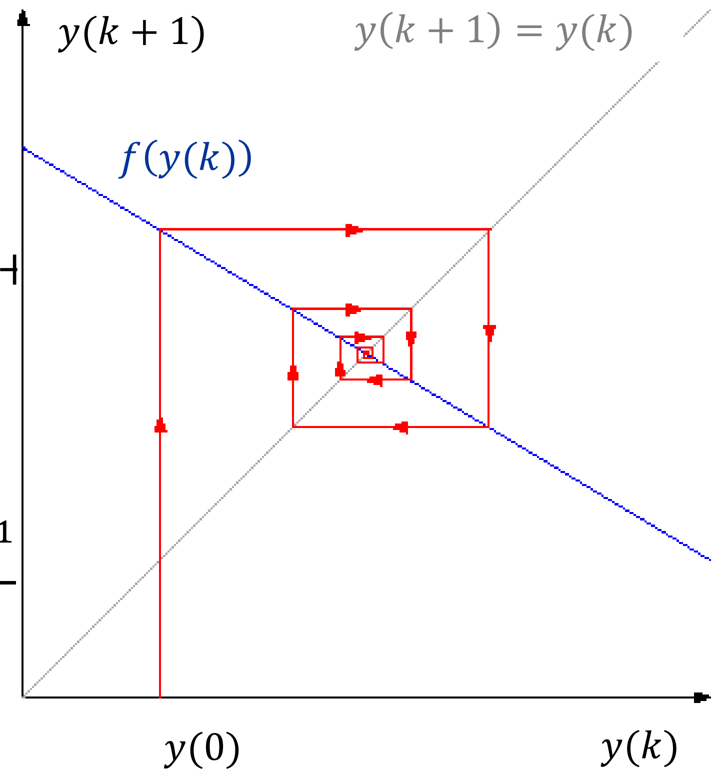
$$a^k(y_0 - y^*) + y^*, y^* = \frac{b}{1-a}, a \neq 1$$

- **Example**

$$y(k+1) = -0.6y(k) +$$

- $y^* = \frac{b}{1-a} = \frac{8}{1+0.6} = 5$
- $y(k) = \left(-\frac{3}{5}\right)^k (2-5) + 5 =$

$$\frac{(-1)^{k+1}3^{k+1}}{5^k}$$



Linear Affine Difference Equations

Mathematical Modelling and Simulation

$$y(k+1) = f\big(y(k)\big) = ay(k) + b, y(0) = y_0, k = 0,1,2,\ldots$$

- **Solution**

$$y(k) = a^k y_0 + b\frac{1-a^k}{1-a} =$$

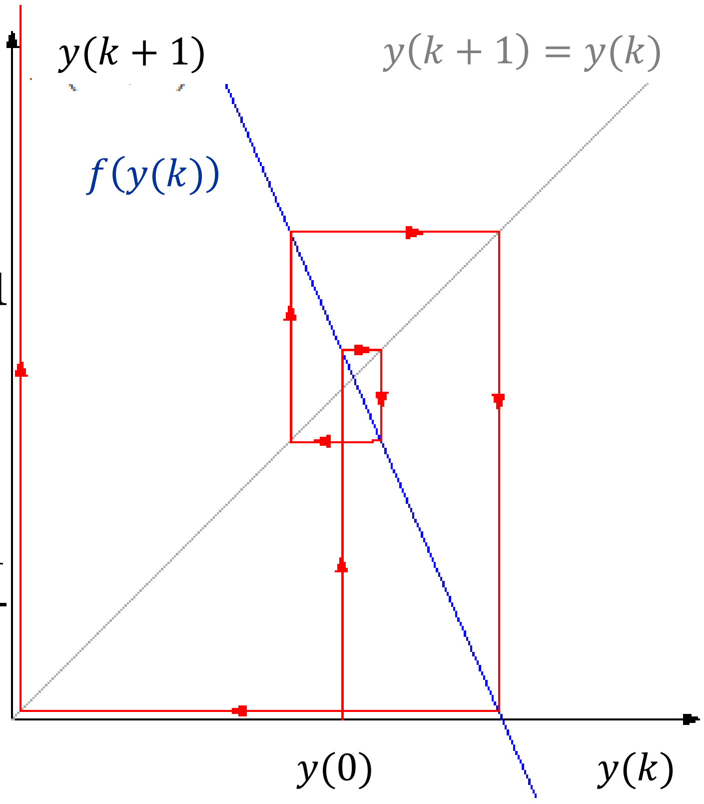
$$a^k(y_0 - y^*) + y^*, y^* = \frac{b}{1-a}, a \neq 1$$

- Example

$$y(k+1) = -2.5y(k) + 1$$

- $y^* = \frac{b}{1-a} = \frac{17.5}{1+2.5} = 5$
- $y(k) = \left(-\frac{5}{2}\right)^k \left(\frac{24}{5} - 5\right) + 5 =$

$$\frac{(-1)^{k+1}5^{k-1}}{2^k}$$



Linear Affine Difference Equations

TU WIEN Mathematical Modelling and Simulation

$$y(k+1) = f\big(y(k)\big) = ay(k) + b, y(0) = y_0, k = 0,1,2, \ldots$$

- **Solution**

$$y(k) = a^k y_0 + b\frac{1-a^k}{1-a} =$$

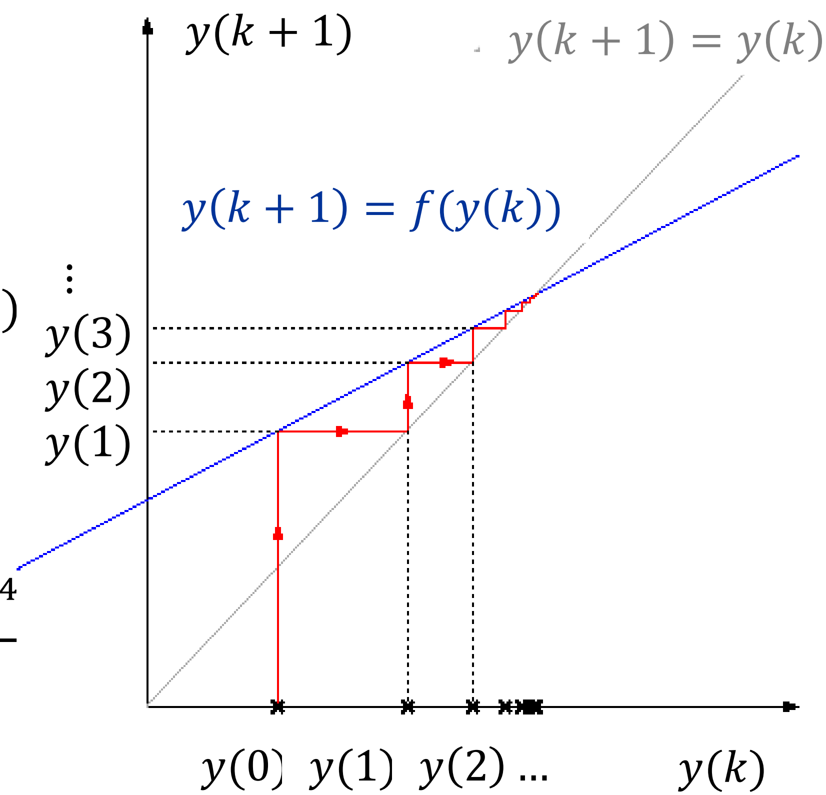
$$a^k(y_0 - y^*) + y^*, y^* = \frac{b}{1-a}, a \neq 1$$

- **Example**

$$y(k+1) = \frac{4}{9}y(k)$$

- $y^* = \frac{b}{1-a} = 6.3$
- $y(k) = \left(-\frac{4}{9}\right)^k \left(\frac{4}{9} - 6.3\right) + 6.3 =$

$$-\frac{2^{2k-2}3^4}{5}$$



Linear Affine Difference Equations

Mathematical Modelling and Simulation

$$y(k+1) = f\big(y(k)\big) = ay(k) + b, y(0) = y_0, k = 0,1,2,\ldots$$

- **Solution**

$$y(k) = a^k y_0 + b\frac{1-a^k}{1-a} =$$
$$a^k(y_0 - y^*) + y^*, y^* = \frac{b}{1-a}, a \neq 1$$

- **Equilibrium – Fixed Point**
  one (or no) fixed point

$$y^* = \frac{b}{1-a}, a \neq 1$$
$$y^* = y_0, a = 1, b = 0$$
$$no\ equilibrium\ for\ a = 1, b \neq 0$$

- **Stability:**

$$stable\ iff\ |a| < 1, \qquad y^*\ attracting$$
$$\text{unstable}\ iff\ |a| \geq 1, \qquad y^*\ repelling$$

Linear Affine Difference Equations

Mathematical Modelling and Simulation

$$y(k+1) = f\big(y(k)\big) = ay(k) + b, y(0) = y_0, k = 0,1,2, \ldots$$

- **Solution**

$$y(k) = a^k y_0 + b\frac{1-a^k}{1-a} =$$
$$a^k(y_0 - y^*) + y^*, y^* = \frac{b}{1-a}, a \neq 1$$

- Classification of Solutions
  Typ of solution depends on a, b and y_0

Main classification

1. $a > 1$
2. $a = 1$
3. $0 < a < 1$
4. $-1 < a < 0$
5. $a = -1$
6. $a < -1$

Sub-classification

1. $y_0 = \frac{b}{1-a}$
2. $y_0 > \frac{b}{1-a}$
3. $y_0 < \frac{b}{1-a}$

	Parameters	Solution Type
1	$a > 1, y_0 = y^*$	Constant
2	$a > 1, y_0 > y^*$	Exponentially increasing without bound
3	$a > 1, y_0 < y^*$	Exponentially decreasing without bound
4	$a = 1, \; b = 0$	Constant
5	$a = 1, \; b > 0$	Linearly increasing without bound
6	$a = 1, \; b < 0$	Linearly decreasing without bound
7	$0 < a < 1, y_0 = y^*$	Constant
8	$0 < a < 1, y_0 > y^*$	Exponentially decreasing to a bound
9	$0 < a < 1, y_0 < y^*$	Exponentially increasing to a bound
10	$-1 < a < 0, y_0 = y^*$	Constant
11	$-1 < a < 0, y_0 > y^*$	Oscillating with decreasing amplitude
12	$-1 < a < 0, y_0 < y^*$	Oscillating with decreasing amplitude
13	$a = -1, y_0 = b/2$	Constant
14	$a = -1, y_0 > b/2$	Oscillating with constant amplitude
15	$a = -1, y_0 < b/2$	Oscillating with constant amplitude
16	$a < -1, y_0 = y^*$	Constant
17	$a < -1, y_0 > y^*$	Oscillating with increasing amplitude
18	$a < -1, y_0 < y^*$	Oscillating with increasing amplitude

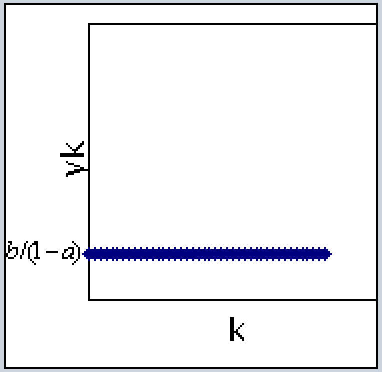
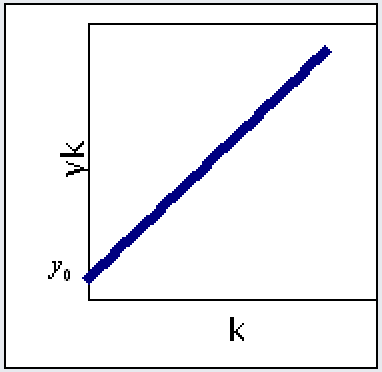
Linear Affine Difference Equations

$$y(k+1) = ay(k) + b$$
$$y(0) = y_0$$
$$y^* = \frac{b}{1-a}, a \neq 1$$

Linear Affine Difference Equations - Classification of Solutions

Mathematical Modelling and Simulation

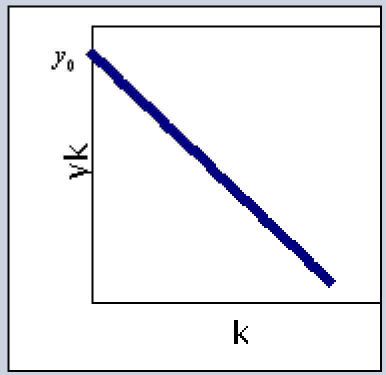
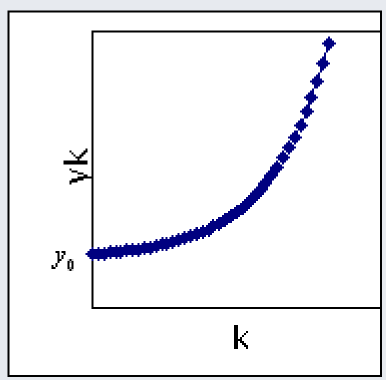
$$y(k+1) = ay(k) + b, y(0) = y_0, y^* = \frac{b}{1-a}, a \neq 1$$

No	Solution Type	Solution Sketch	Parameters
1	Constant		$a > 1, y_0 = y^*$ $a = 1, \ b = 0$ $0 < a < 1, y_0 = y^*$ $-1 < a < 0, y_0 = y^*$ $a = -1, y_0 = b/2$ $a < -1, y_0 = y^*$
2	Linearly increasing without bound		$a = 1, \ b > 0$

Linear Affine Difference Equations - Classification of Solutions

Mathematical Modelling and Simulation

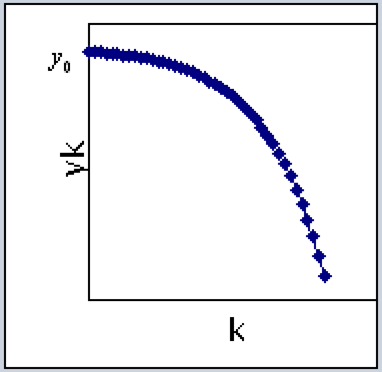
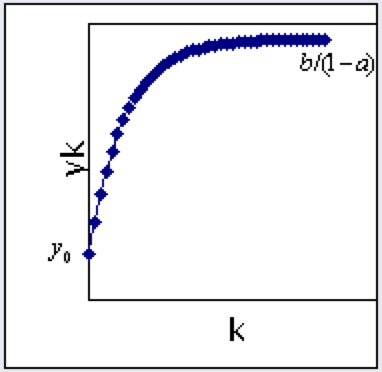
$$y(k+1) = ay(k) + b, y(0) = y_0, y^* = \frac{b}{1-a}, a \neq 1$$

No	Solution Type	Solution Sketch	Parameters
3	Linearly decreasing without bound		$a = 1,\ b < 0$
4	Exponentially increasing without bound		$a > 1, y_0 > y^*$

Linear Affine Difference Equations - Classification of Solutions

Mathematical Modelling and Simulation

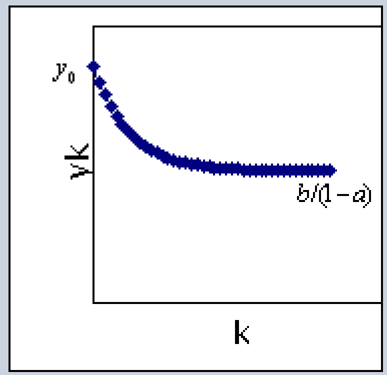
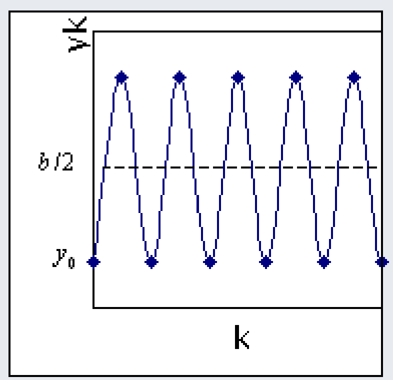
$$y(k+1) = ay(k) + b, y(0) = y_0, y^* = \frac{b}{1-a}, a \neq 1$$

No	Solution Type	Solution Sketch	Parameters
5	Exponentially decreasing without bound		$a > 1, y_0 < y^*$
6	Exponentially increasing to a bound		$0 < a < 1, y_0 < y^*$

Linear Affine Difference Equations - Classification of Solutions

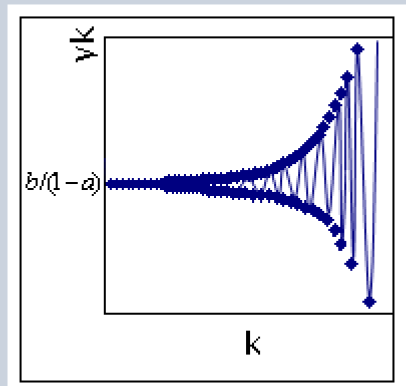
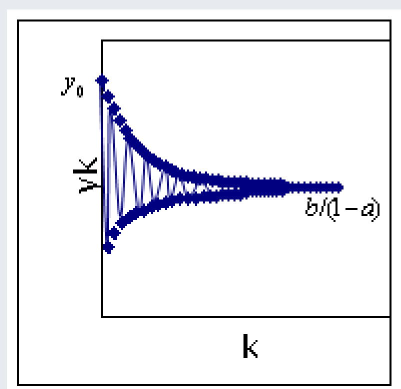
Mathematical Modelling and Simulation

$$y(k+1) = ay(k) + b, y(0) = y_0, y^* = \frac{b}{1-a}, a \neq 1$$

No	Solution Type	Solution Sketch	Parameters
7	Exponentially decreasing to a bound		$0 < a < 1, y_0 > y^*$
8	Oscillating with constant amplitude		$a = -1, y_0 > b/2$ $a = -1, y_0 < b/2$

Linear Affine Difference Equations - Classification of Solutions

$$y(k+1) = ay(k) + b, y(0) = y_0, y^* = \frac{b}{1-a}, a \neq 1$$

No	Solution Type	Solution Sketch	Parameters
9	Oscillating with increasing amplitude		$a < -1, y_0 > y^*$ $a < -1, y_0 < y^*$
10	Oscillating with decreasing amplitude		$-1 < a < 0, y_0 > y^*$ $-1 < a < 0, y_0 < y^*$

Applications to finance

- Actual balance $y(n)$ after n compounding periods with annual interest I, compounded m times a year and constant amount b added at the end of every compounding period:

$$y(k+1) = \left(1 + \frac{I}{m}\right) y(k) + b$$

Solution:

$$y^* = \frac{b}{1-a} = \frac{mb}{I}\ , y(k) = \left(1 + \frac{I}{m}\right)^k \left(y_0 - \frac{mb}{I}\right) + \frac{mb}{I}$$

Applications to economics

- Supply and Demand
  - $S(n)$, $D(n)$, $P(n)$ ... supply, demand, price in the year n
  - Set of assumptions:

- $S(k+1) = sP(k) + a, a > 0$ s sensitivity of producers to price
- $D(k+1) = -dP(k+1) + b$ d sensitivity of consumers to price
- $S(k+1) = D(k+1)$ via adjustment of price/bargaining

first order affine dynamical system

$\rightarrow -dP(k+1) + b = sP(k) + a$

$$\rightarrow P(n+1) = -\frac{s}{d}P(n) + \frac{(b-a)}{d}, P^* = \frac{b-a}{d+s}$$

Applications to economics

Mathematical Modelling and Simulation

- Supply and Demand
  - $S(n)$, $D(n)$, $P(n)$ ... supply, demand, price in the year n
  - Set of assumptions:

- $S(k+1) = sP(k) + a, a > 0$ s sensitivity of producers to price
- $D(k+1) = -dP(k+1) + b$ d sensitivity of consumers to price
- $S(k+1) = D(k+1)$ via adjustment of price/bargaining

first order affine dynamical system

$$\rightarrow -dP(k+1) + b = sP(k) + a$$

$$\rightarrow P(n+1) = -\frac{s}{d}P(n) + \frac{(b-a)}{d}, P^* = \frac{b-a}{d+s}$$

Applications to economics

- Supply and Demand
  - $S(n)$, $D(n)$, $P(n)$ ... supply, demand, price in the year n

- $P(k+1) = \frac{s}{d}P(k) + \frac{b-a}{d}$

first order affine dynamical system

- Fixed Point: $P^* = \frac{b-a}{s+d}$

- General Solution:

$$P(k) = c\left(-\frac{s}{d}\right)^k + p$$

stable for

$$-1 < -\frac{s}{d} < 1$$

Cobweb theorem of economics

Mathematical Modelling and Simulation

Difference Equations with MATLAB

Case Study: Logistic Equation

Repetition: Difference Equation

Mathematical Modelling and Simulation

- Problems defined by

$$x_{n+1} = f(n, x_n, x_{n-1}, \dots, x_{n-d})$$
$$x_0 = k$$

  are called difference-equations.
- Solution of these equations is given by a sequence of, probably vector-valued, numbers x_n with a certain initial value k.

Repetition: Connection between Difference E. and Differential E.

- $x_{n+1} = f(n, x_n, x_{n-1}, \dots, x_{n-d}) \Rightarrow$
$$x_{n+1} - x_n = g(n, x_n, x_{n-1}, \dots, x_{n-d})$$

Difference!

Solutions of difference equations are gained by the <u>sum of all differences</u> starting at a specific value!

Solutions of differential equations are gained by the <u>sum of all infinitesmial differencials</u> starting at a specific value! In this case, the sum is called <u>integral</u>!

Repetition: Connection between Difference E. and Differential E.

Mathematical Modelling and Simulation

A solution of a difference equation is a sequence. We receive a value for each interation step!
$\{0,1,2,\ldots,n\}$
This is usually called explicit representation of the sequence in contrast to a recursive one.

A solution of a differential equation is a „very infinite“ sequence“. We receive a value for **each** timepoint
$[0, t_{end}]$
Those kind of „sequences“ are usually called **functions**!

Repetition: Connection between Difference E. and Differential E.

Mathematical Modelling and Simulation

We differ between linear and nonlinear difference equations. E.g.:
Linear: $x_{n+1} = 4x_n + 2$
Nonlinear: $x_{n+1} = x_n^2 + x_n$

We differ between linear and nonlinear differentiale equations. E.g.:
Linear: $x' = 3x + 2$
Nonlinear: $x' = x^2$

Repetition: Connection between Difference E. and Differential E.

We can perform a z-Transformation

$$x_{n+1} - x_n = 3x_n + 2$$
$$a(z) = \frac{2}{\frac{1}{z} - 3}$$

We can perform a Laplace-Transformation

$$x' = 3x + 2$$
$$\mathrm{t}(s) = \frac{2}{\frac{1}{s} - 3}$$

Repetition: Connection between Difference E. and Differential E.

Mathematical Modelling and Simulation

Finding a explicit solution is usually very tricky! Sometimes comparisons with geometric sequences can lead to sucess.

Anyway values can be calculated directly through the recursive formula.

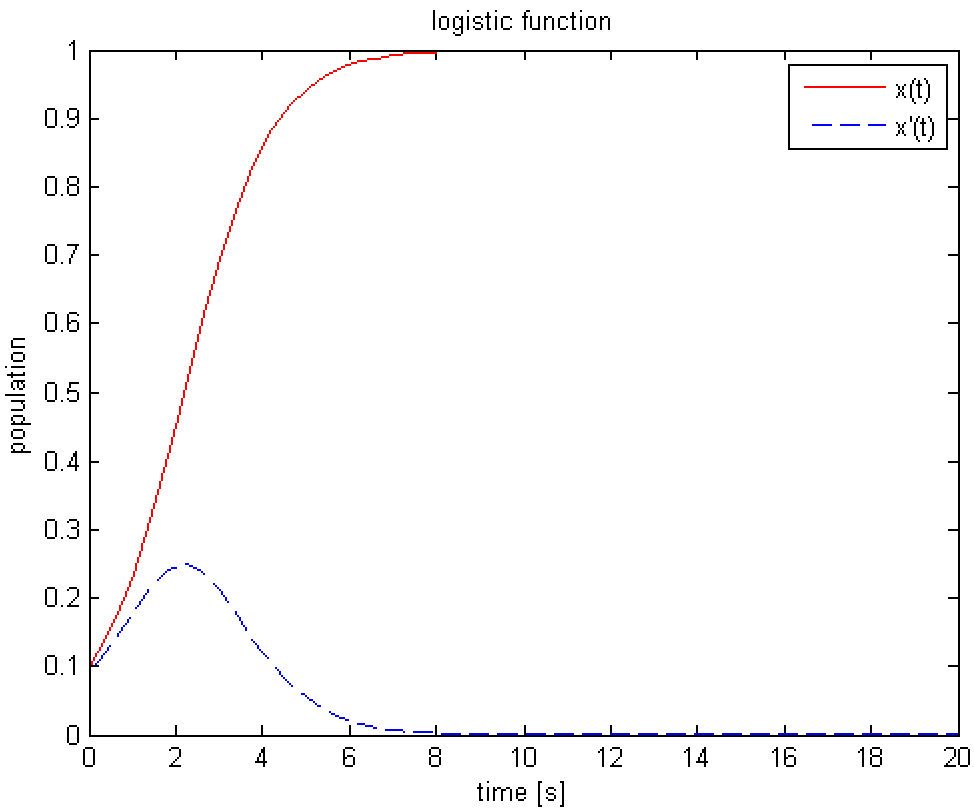
Finding an analytic solution can be performed with analytical methods. If no solutions can be found this way a numerical approximation method needs to be used usually leading to difference equations.

Repetition: Comparison Logistic Difference Equation and Logistic Differential Equation

- Logistic differential equation is given by
$$x' = ax(b - x)$$
- The corresponding logistic-difference equation is given by
$$x_{n+1} = x_n + ax_n(b - x_n)$$

Repetition: Comparison Logistic Difference Equation and Logistic Differential Equation

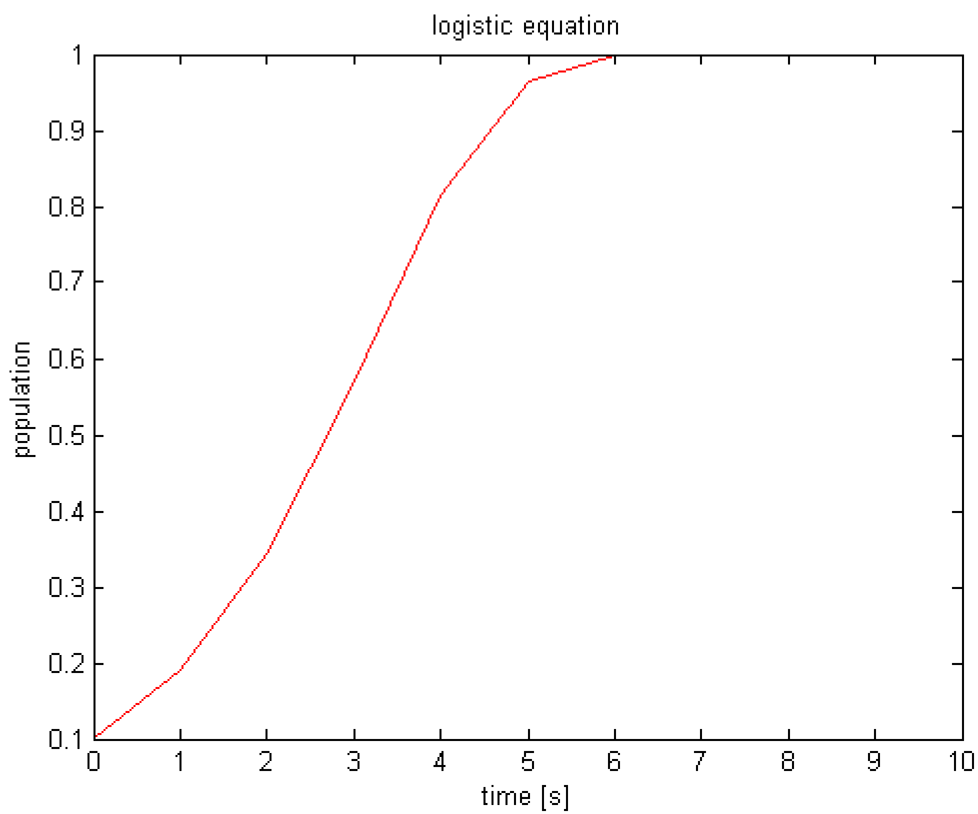
Solutions of the logistic differential equation are steady, and behave similar for all parameters.



Repetition: Comparison Logistic Difference Equation and Logistic Differential Equation

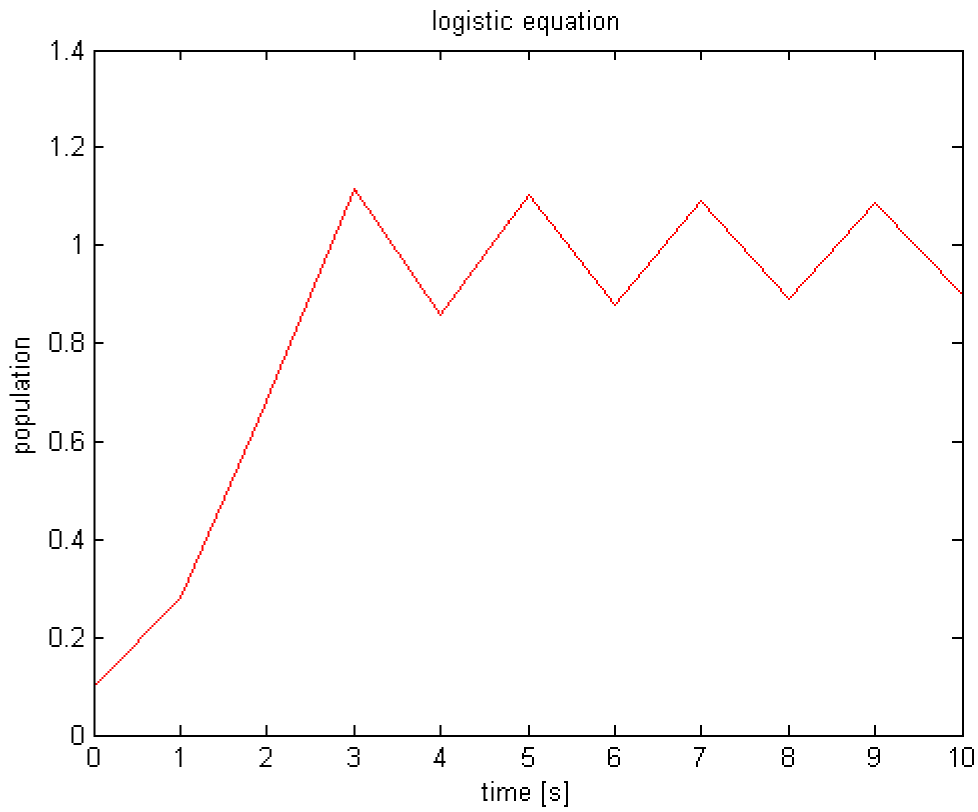
Mathematical Modelling and Simulation

Solutions of the logistic difference equation are unsteady and seem to differ extremely for different parameters



Repetition: Comparison Logistic Difference Equation and Logistic Differential Equation

Solutions of the logistic difference equation are unsteady and seem to differ extremely for different parameters



Conclusion

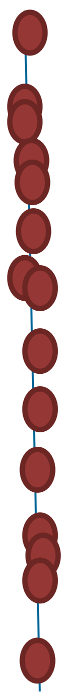
Mathematical Modelling and Simulation

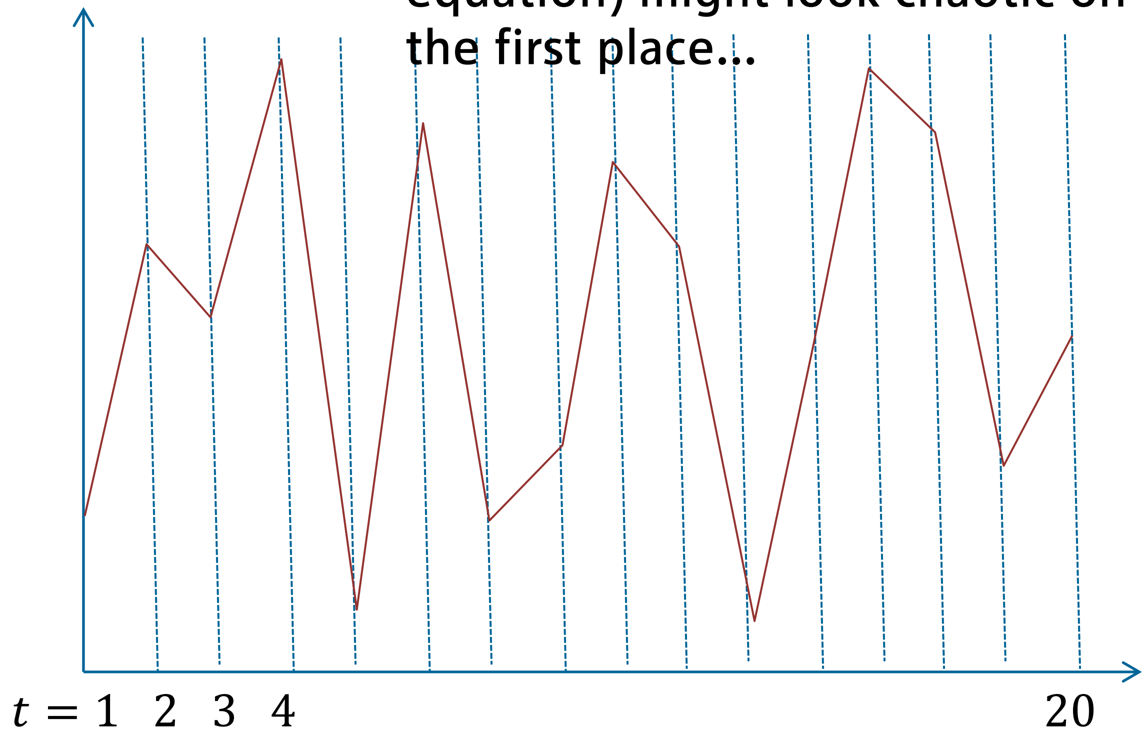
difference equations are a lot more than just discrete versions of differential equations!

Logistic Equation and the Border to Chaos

- What is an accumulation point?

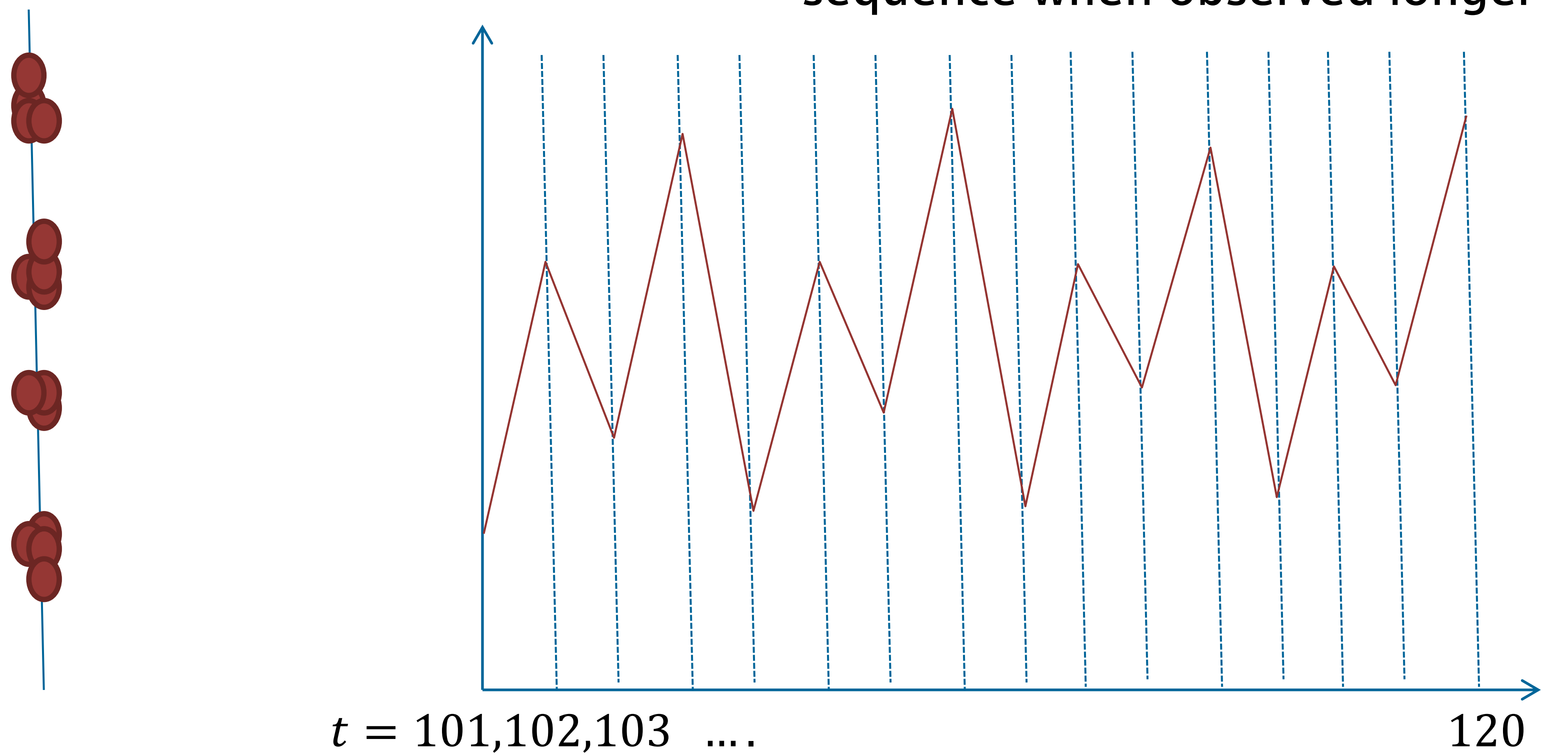
Although a sequence (i.e. a solution of a difference equation) might look chaotic on the first place...





- **What is an accumulation point?**

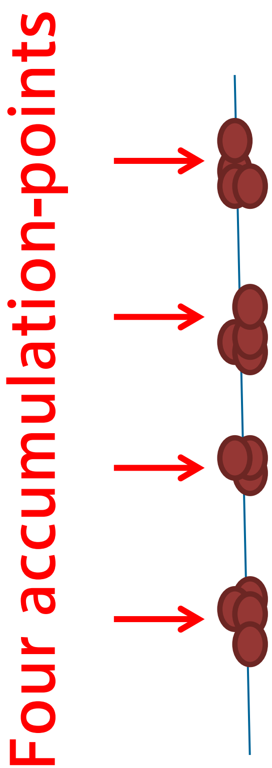
… one might observe a „convergence“ to a periodic sub-sequence when observed longer



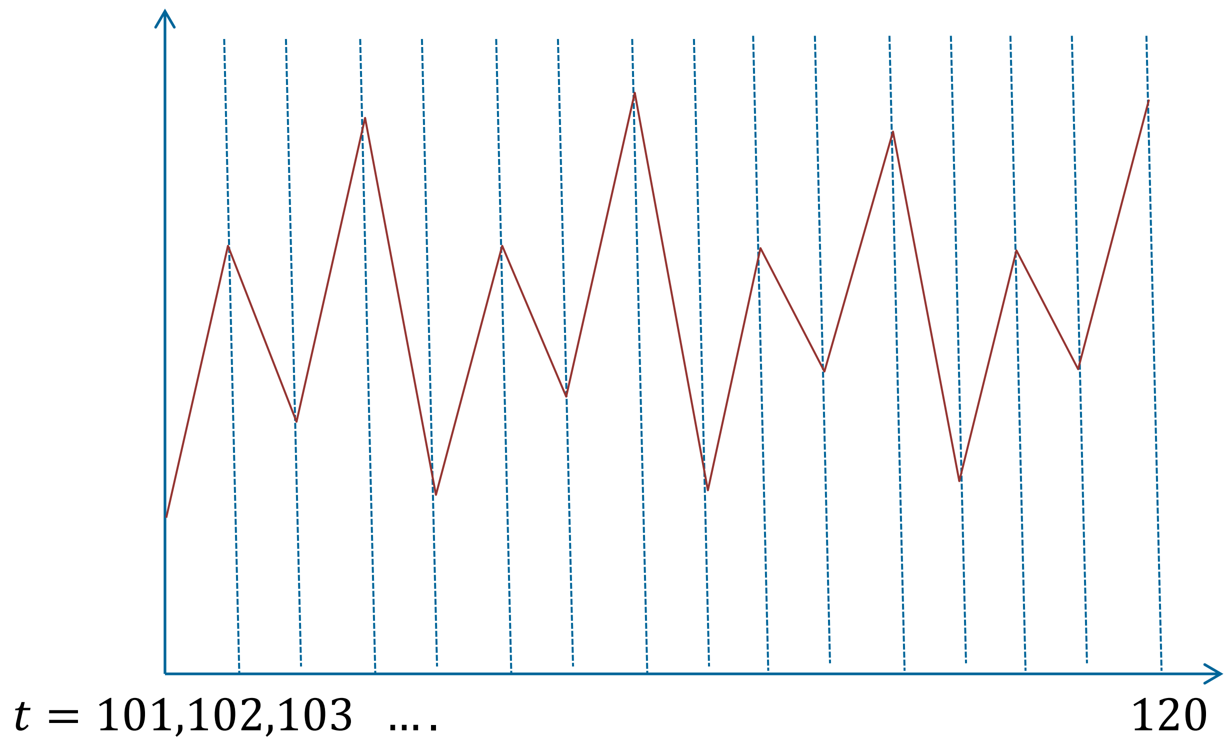
Logistic Equation and the Border to Chaos

Mathematical Modelling and Simulation

- What is an accumulation point?



... one might observe a „convergence“ to a periodic sub-sequence when observed longer



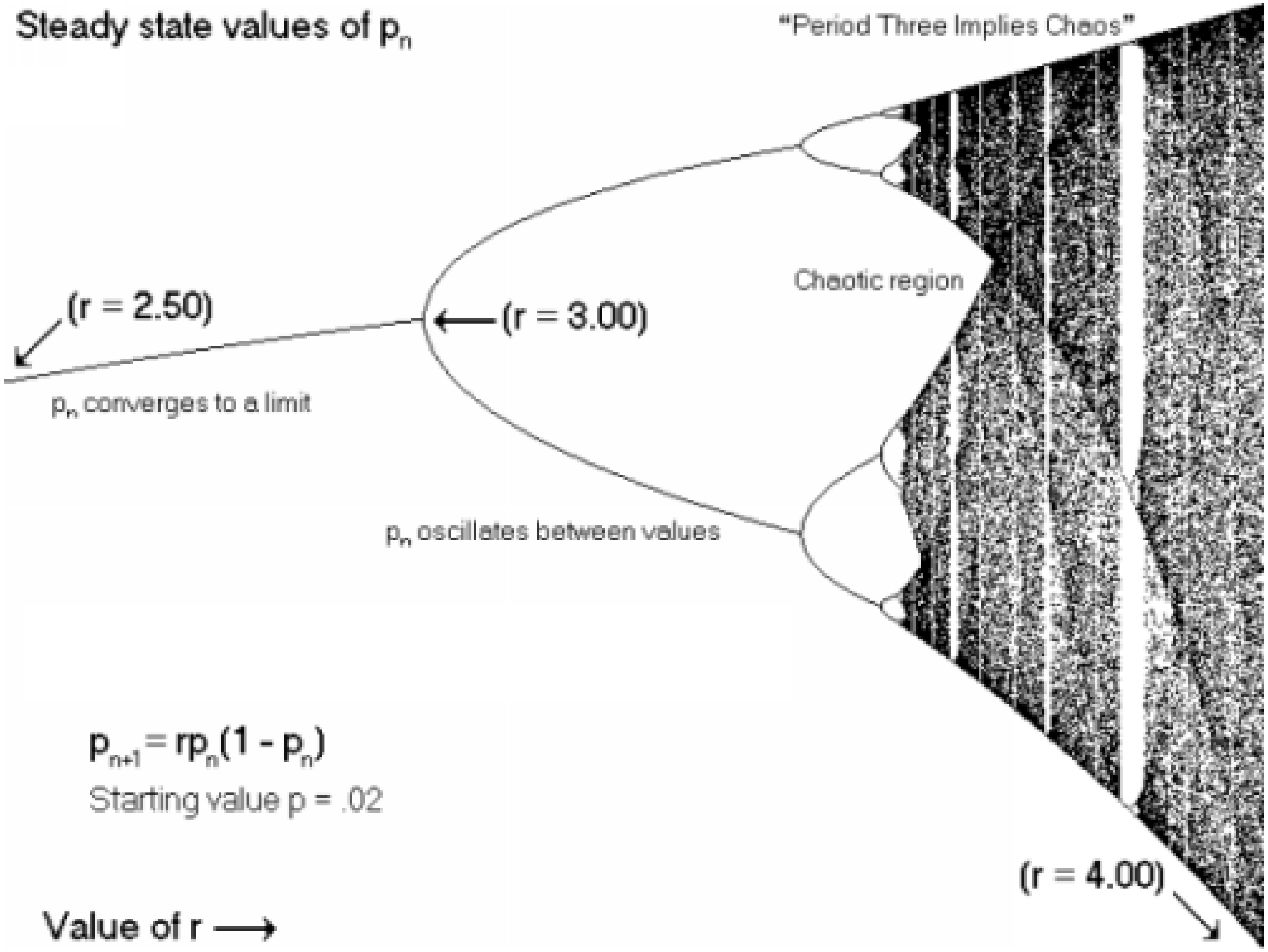
$$x_{n+1} = p x_n (1 - x_n)$$

Experiments with MATLAB/Simulink

Mathematical Modelling and Simulation

How many accumulation points??

Bifurcation



Mathematical Modelling and Simulation

- A couple (person) saved some money planning to have a child

Case Study: „Baby Planner“
Problem Definition

- A couple (person) saved some money planning to have a child
- Net income of couple after the birth is fixed 1700€/month already added financial benefits related to the child. They receive the money at the end of the month.

Case Study: „Baby Planner“
Problem Definition

Mathematical Modelling and Simulation

- A couple (person) saved some money planning to have a child
- Net income of couple after the birth is fixed 1700€/month already added financial benefits related to the child. They receive the money at the end of the month.
- Fixed costs (flat, insurance, car..) after birth is approximated to 1150€/month which they have to pay after the second week of each month

Case Study: „Baby Planner“
Problem Definition

Mathematical Modelling and Simulation

- A couple (person) saved some money planning to have a child
- Net income of couple after the birth is fixed 1700€/month already added financial benefits related to the child. They receive the money at the end of the month.
- Fixed costs (flat, insurance, car..) after birth is approximated to 1150€/month which they have to pay after the second week of each month
- Costs per week after birth are approximated with 150€.

Case Study: „Baby Planner"
Problem Definition

Mathematical Modelling and Simulation

- A couple (person) saved some money planning to have a child
- Net income of couple after the birth is fixed 1700€/month already added financial benefits related to the child. They receive the money at the end of the month.
- Fixed costs (flat, insurance, car..) after birth is approximated to 1150€/month which they have to pay after the second week of each month
- Costs per week after birth are approximated with 150€.
- Income of the couple is saved with interest rate of 0.1%/month.

Research Question:

Does the money last for 18 years?

Difference Equation Model

Mathematical Modelling and Simulation

- We observe that the type of the recursion depends on the division of the index by 4
- $x_{n+1} = x_n - 150$, if $n \equiv 1(4)$ or $n \equiv 3(4)$
- $x_{n+1} = x_n - 150 - 1150$, if $n \equiv 2(4)$
- $x_{n+1} = (x_n - 150) \cdot (1 + \frac{0.1}{100}) + 1700$, if $n \equiv 0(4)$

Mathematical Modelling and Simulation

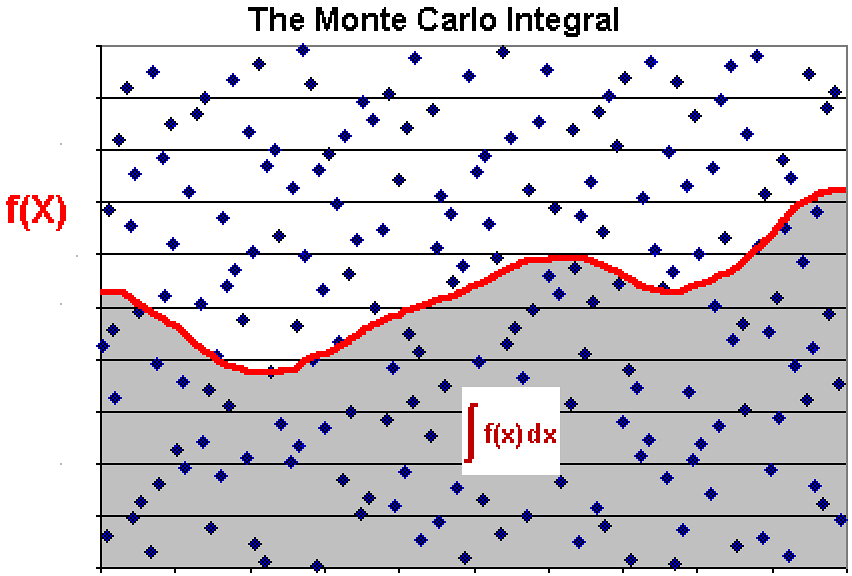
Implementation in Simulink

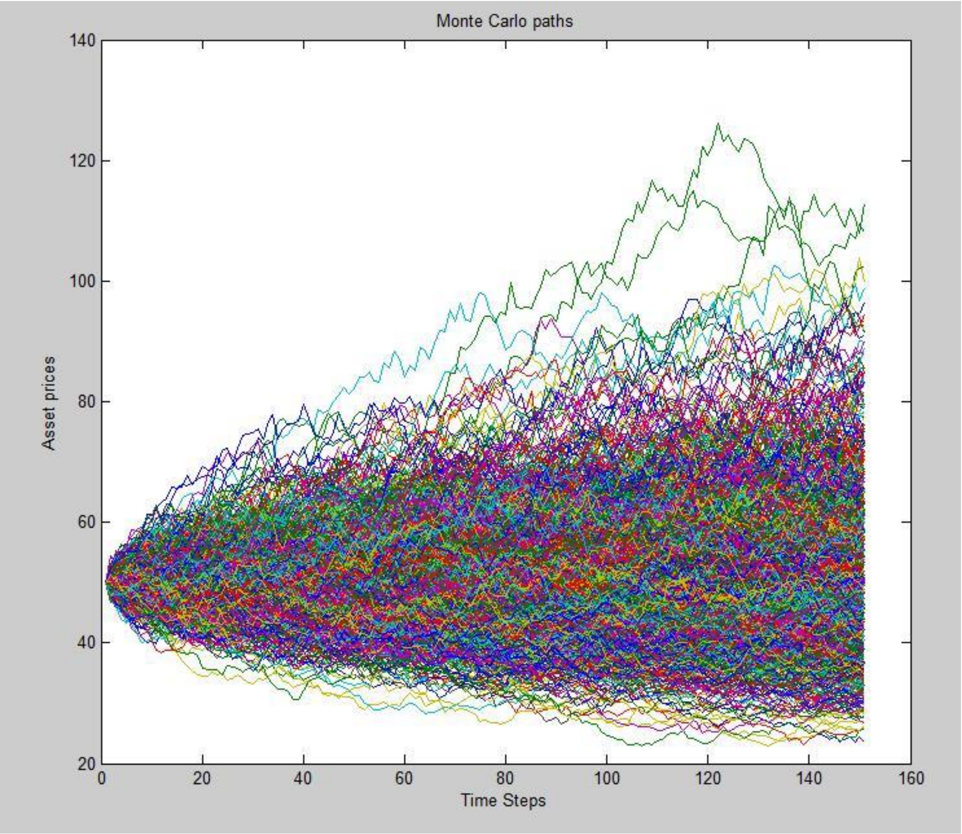
Adaption of the Model

- Unfortunately the anount of money spent each week is not known perfectly.
- We introduce a random variable making the simulation stochastic. This raises new questions:

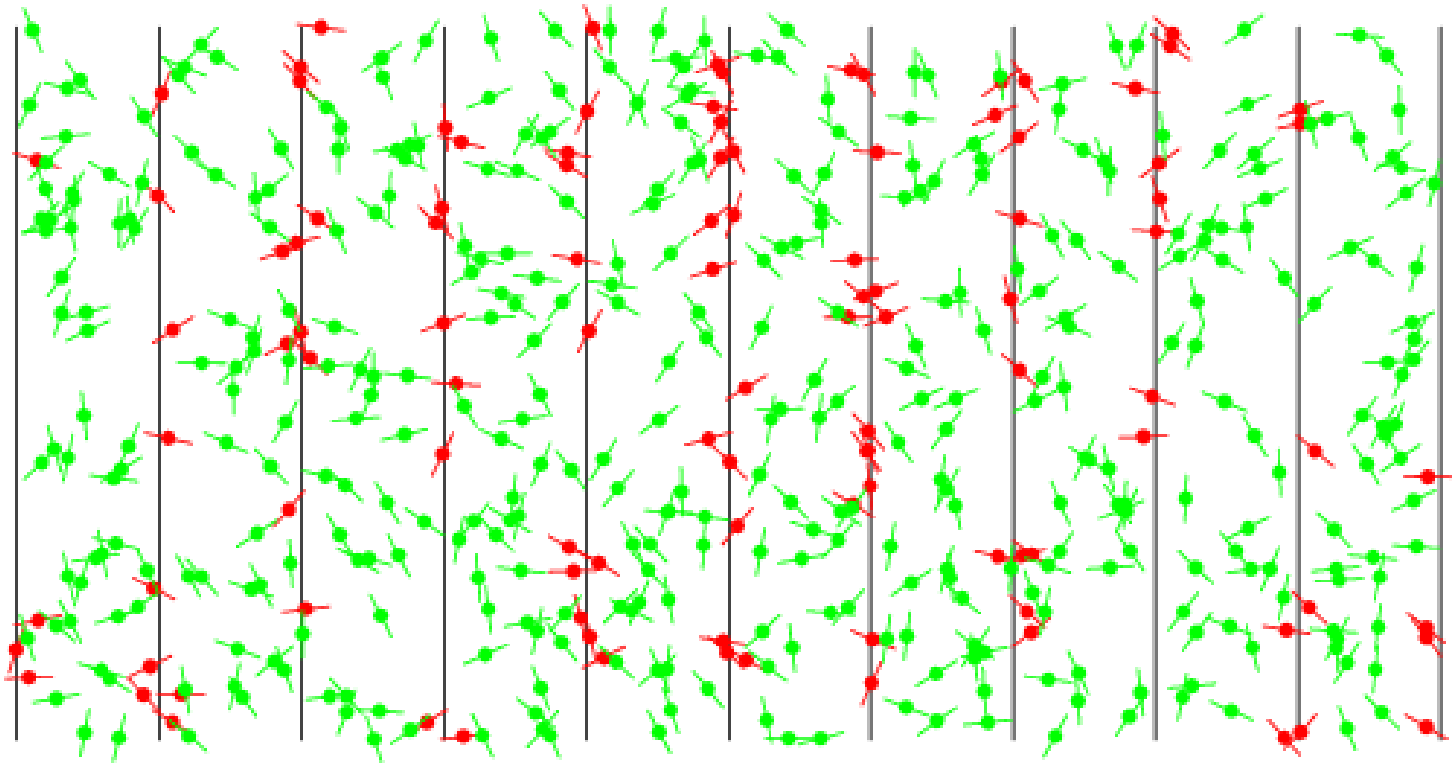
Can I expect that the money will last for 18 years?
How confident is this assumption?
Variance? Mean? Quantiles?

Monte Carlo Simulation





Buffon's Needle Problem



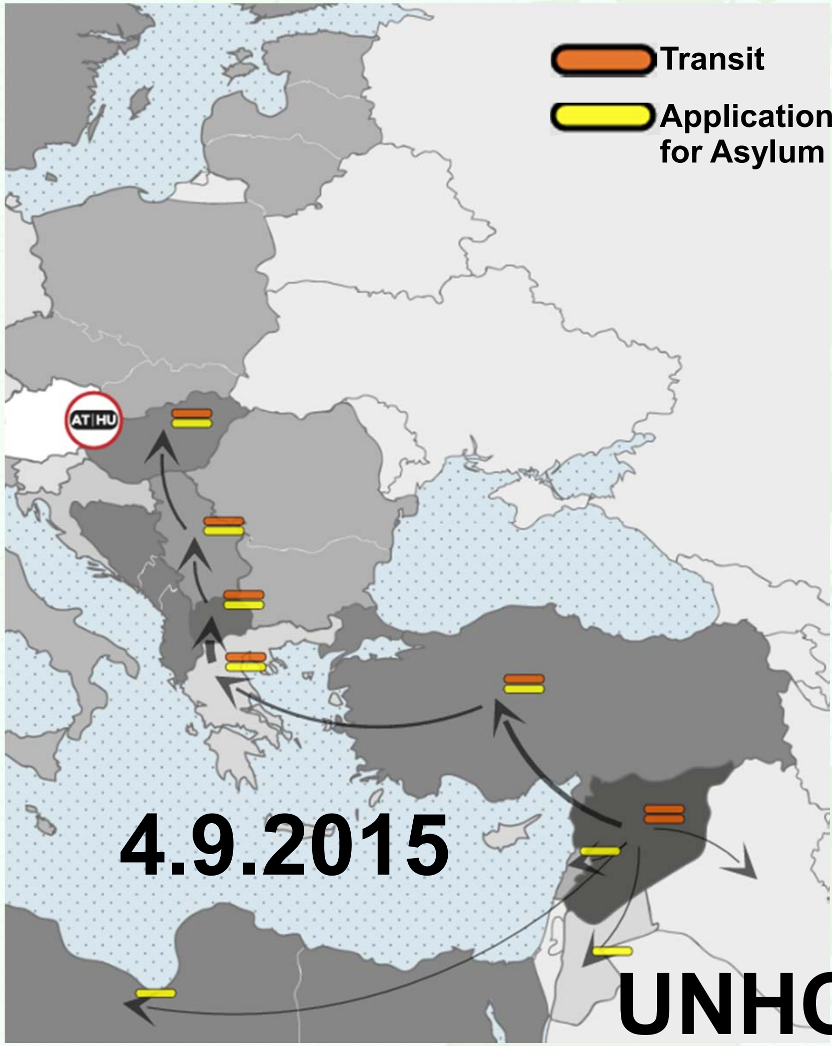
Migration Analysis by Modelling and Simulation

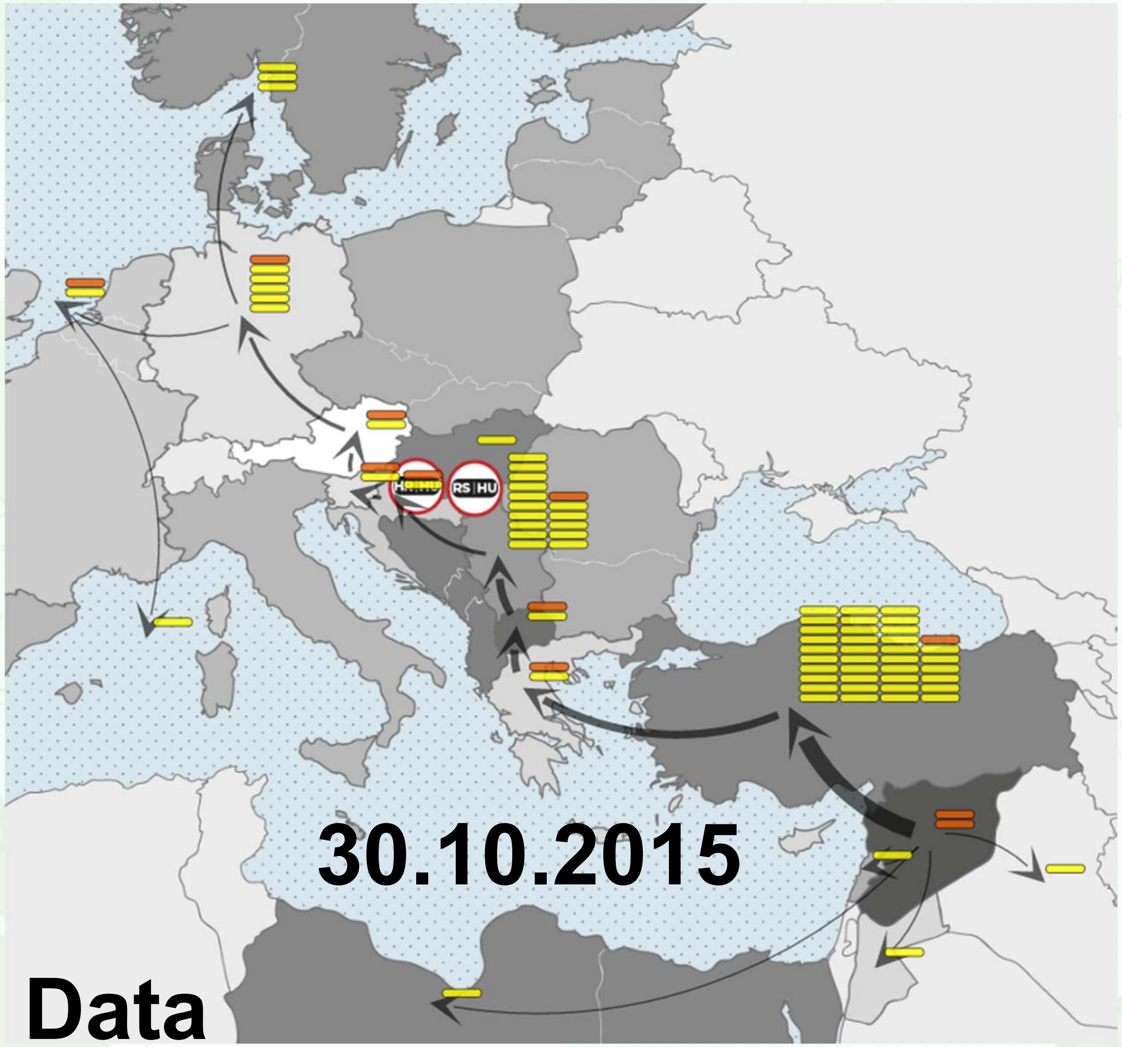
Felix Breitenecker[1], Tamara Vobruba[1],
Andreas Körner[1], Nikolas Popper[1,2]

[1]TU Wien, COCOS - Computational Complex Systems and ARGESIM/Mathematical Modelling and Simulation
[2] dwh Simulation Services

TU WIEN TECHNISCHE UNIVERSITÄT WIEN Vienna | Austria

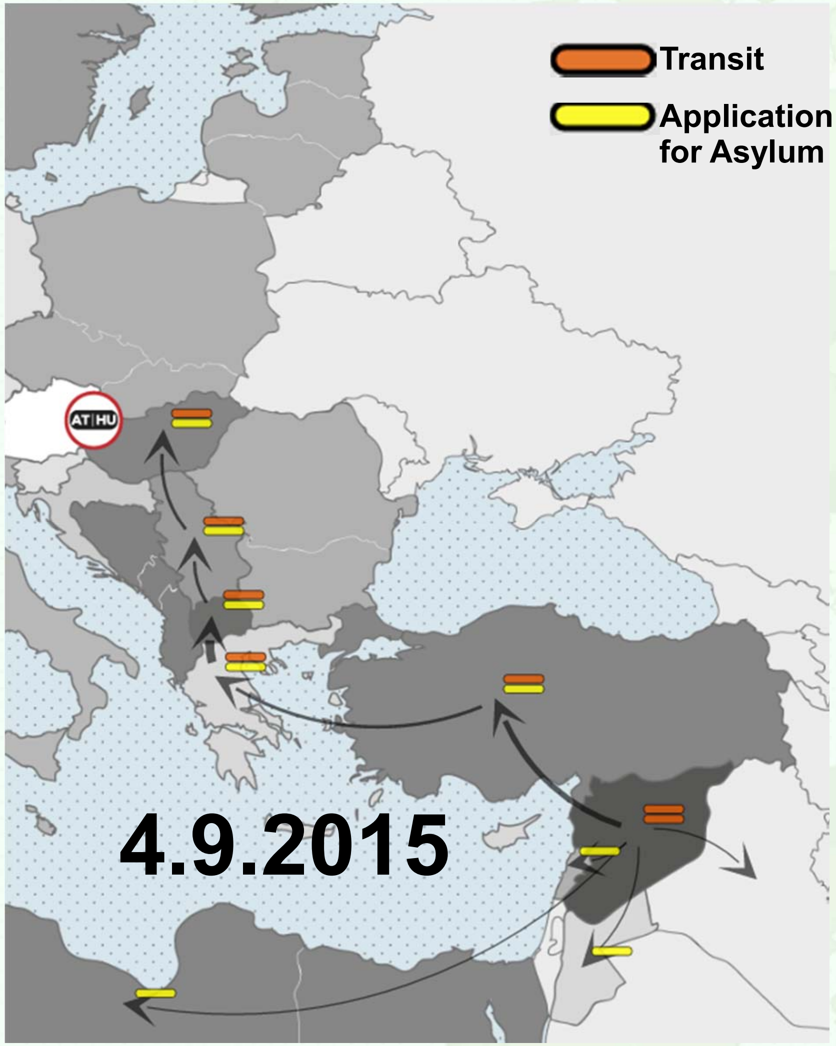
Refugee Crisis 2015 1.9.2015 – 30.10.2015

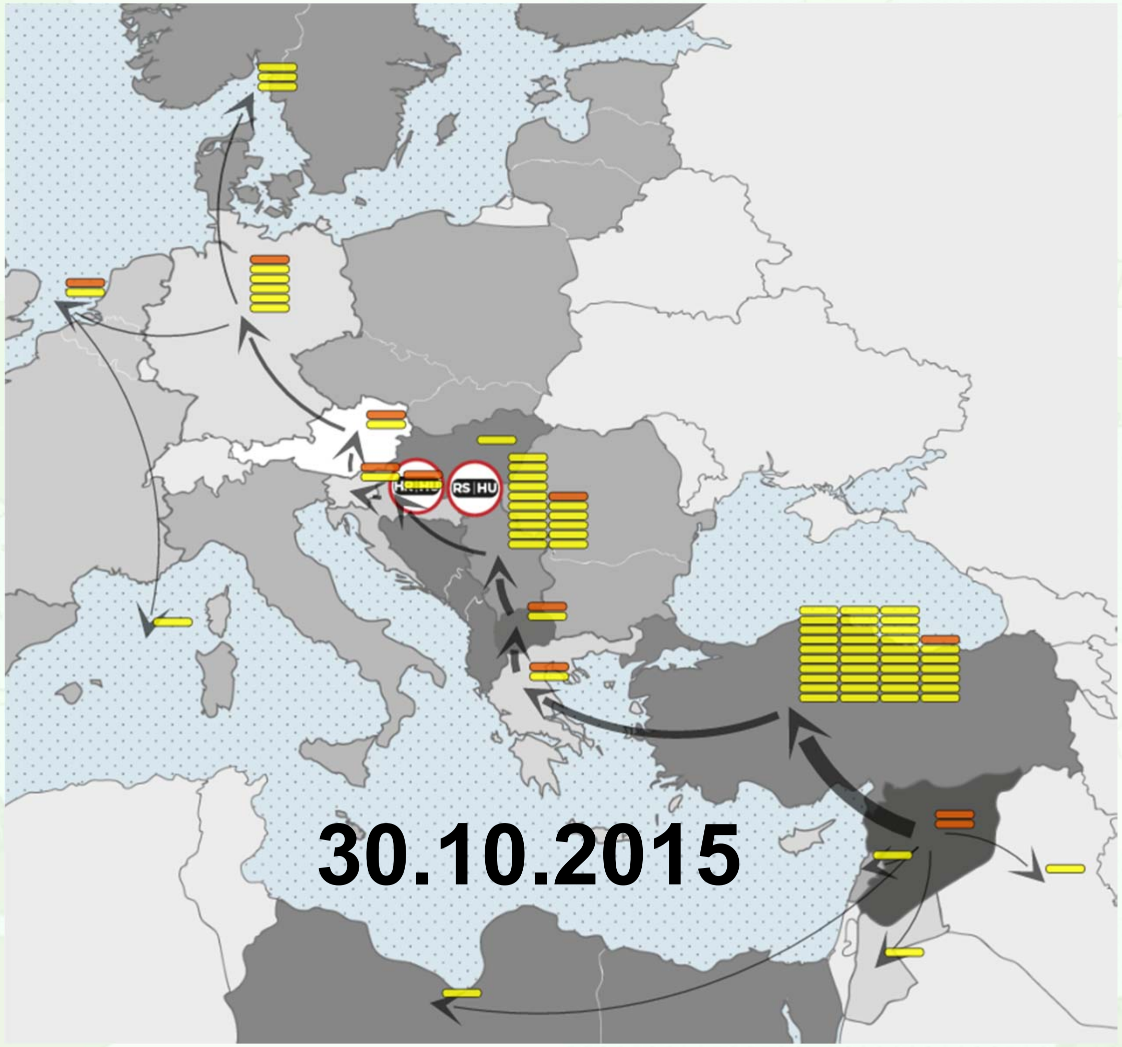




UNHCR Data

Refugee Crisis 2015 1.9.2015 – 30.10.2015

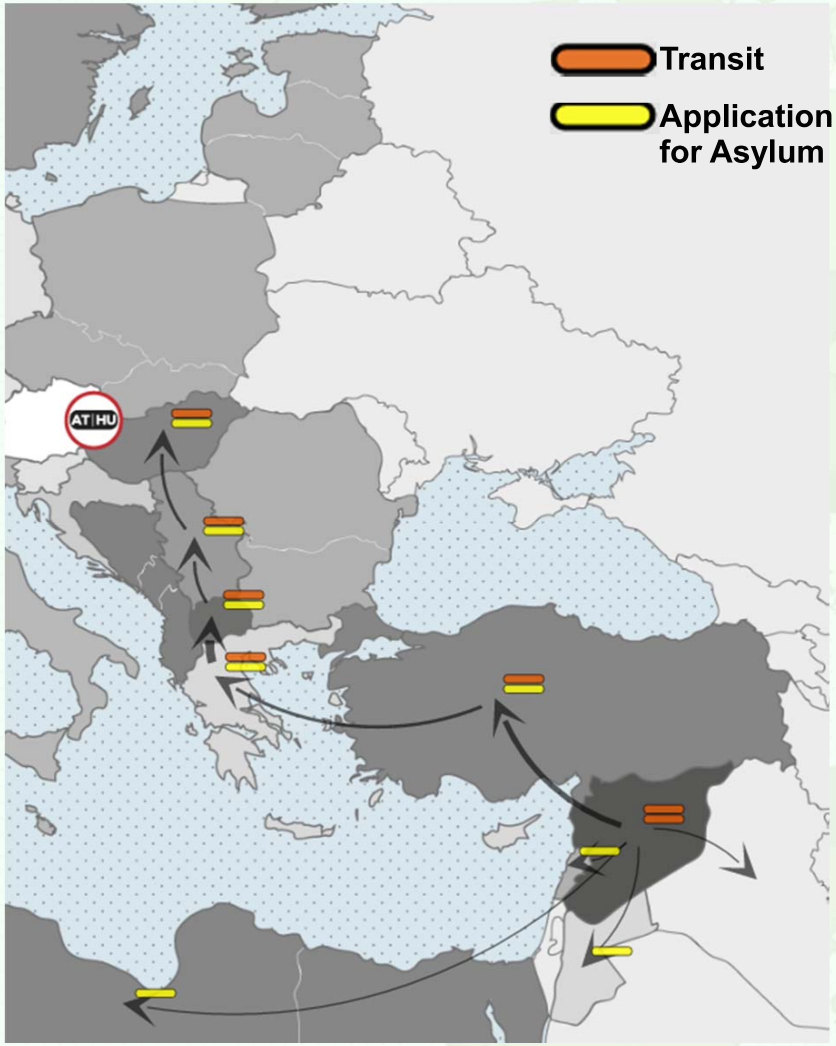




Modelling & Simulation ?

TU WIEN Mathematical Modelling and Simulation

Refugee Crisis 2015 1.9.2015 – 30.10.2015



- Spatial Interaction Model
- Social Gravity Model
- Migration Model

Modelling & Simulation ?

Spatial Interaction Model

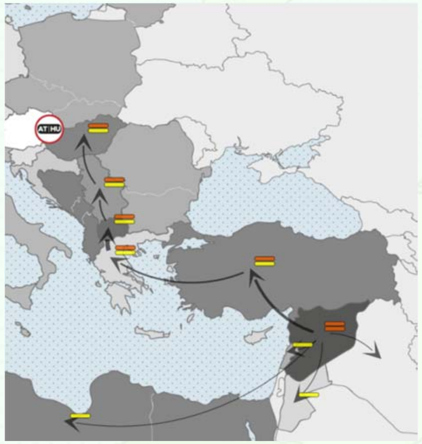
- Spatial interaction = transmission/movement over space resulting of decision making process

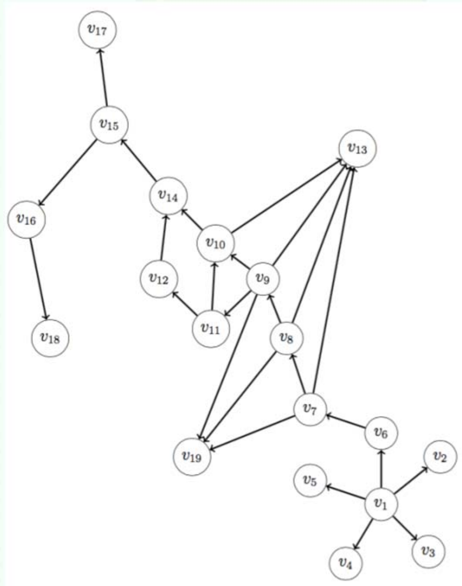
- Decision making process realised by relation of influencing factors

- Applications: flow of traffic, commuters, migrants, goods or messages,...

- Interactions between regions/populations

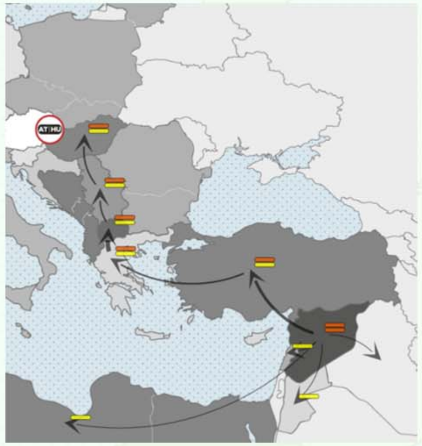
- Regions/populations represented trough a directed graph



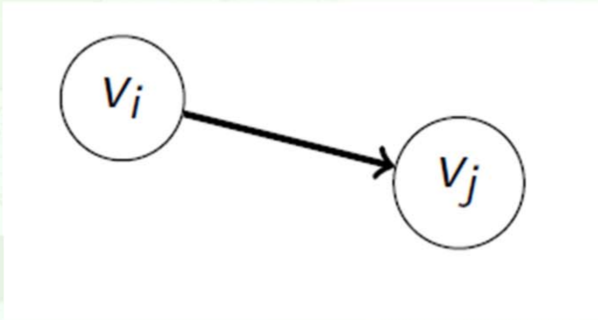


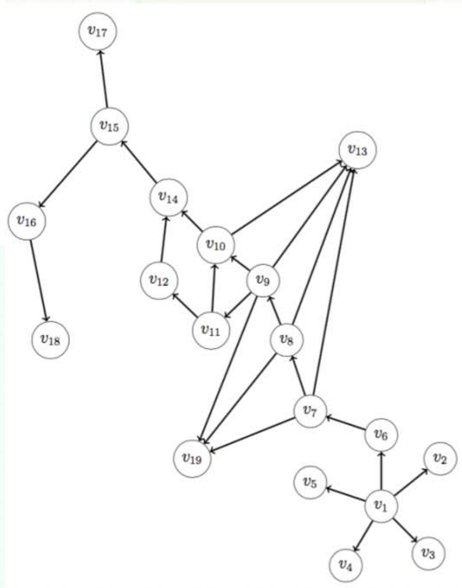
Spatial Interaction Model

- Regions/populations represented trough a directed graph
- Approach for describing any kind of interaction $I_{i,j}$ between regions or populations v_i and v_j



$$I_{i,j} = f\left(R_i, A_j, C_{i,j}\right)$$





R_i Repulsion attributes in v_i

A_j Attraction attributes in v_j

$C_{i,j}$ separation attributes between v_i and v_j

$$I_{i,j} = f(R_i, A_j, C_{i,j}) \quad ?$$

Gravity Model

- Specific form of Spatial Interaction Model
- Social physics: analogies between social behaviour and physics
- Relation of interaction based on law of gravity
- Long history in social sciences:
  1924 Ernest Young: movement of farm migration

$$M = k\,\frac{F}{D^2}$$

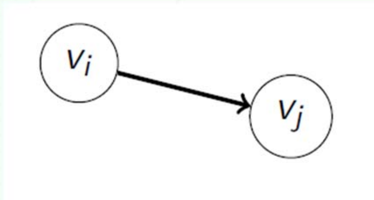
M Migration

F Intensity of attraction

D Distance

k proportional constant

$$I_{i,j} = f\left(R_i, A_j, C_{i,j}\right) \quad ?$$



Gravity Model: Development

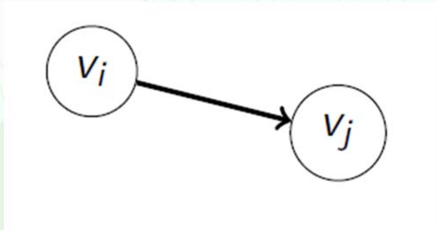
1941 John Steward: concept of demographic gravitation

$$I_{i,j} = G\frac{P_i \cdot P_j}{d_{i,j}^2}$$

P_i, P_j population masses (attributes)

G constant, d Distance

$$I_{i,j} = f(R_i, A_j, C_{i,j}) \quad ?$$



1950 John Steward: refined formulation to include heterogeneity of population masses

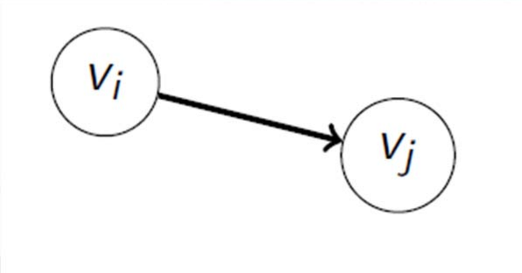
$$I_{i,j} = G\frac{w_i P_i \cdot w_j P_j}{d_{i,j}^2}$$

w_i, w_j weights of population masses

Gravity Model: Model Equations

The class of gravity models for spatial interaction behaviour follows the form

$$I_{i,j} = f\left(R_i, A_j, C_{i,j}\right) \quad ?$$



$$I_{i,j} = R(i) \cdot A(j) \cdot F(i,j)$$

$I_{i,j} \in \mathbb{R}$	interaction between v_i and v_j
$R: \mathbb{R}^N \to \mathbb{R}$	function of repulsive attributes in v_i
$A: \mathbb{R}^M \to \mathbb{R}$	function of attractive attributes in v_j
$F: \mathbb{R}^K \to \mathbb{R}$	function of separation attributes between v_i and v_j (**F** usually non increasing)

Migration Model: Dynamic Equations

- Migrants in v_i at time t $\quad M_i(t)$
- Interaction $\quad I_{i,j}(t) = R(r_i(t)) \cdot A(a_j(t)) \cdot F(c_{i,j}(t))$
- Migrants from v_i to v_j $\quad M_{i,j}(t) = I_{i,j}(t) \cdot M_i(t)$
- Migrants in v_i at time $t+1$ $\quad M_i(t+1) = M_i(t) + M_{i,j}(t)$

Migration Model: Difference Equation

- Migrants in v_i at time t $\quad M_i(t)$
- Interaction $\quad I_{i,j}(t) = R(r_i(t)) \cdot A(a_j(t)) \cdot F(c_{i,j}(t))$
- Migrants from v_i to v_j $\quad M_{i,j}(t) = I_{i,j}(t) \cdot M_i(t)$
- Migrants in v_i at time $t+1$ $\quad M_i(t+1) = M_i(t) + M_{i,j}(t)$

$$M_i(t+1) = M_i(t) + I_{i,j}(t) \cdot M_i(t)$$

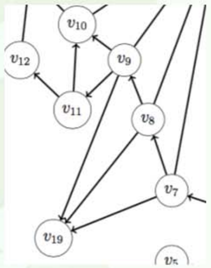
Migration Model: Difference Equation

- Migrants in v_i at time t $\quad M_i(t)$
- Interaction $\quad I_{i,j}(t) = R(r_i(t)) \cdot A(a_j(t)) \cdot F(c_{i,j}(t))$
- Migrants from v_i to v_j $\quad M_{i,j}(t) = I_{i,j}(t) \cdot M_i(t)$
- Migrants in v_i at time $t+1$ $\quad M_i(t+1) = M_i(t) + M_{i,j}(t)$

$$M_i(t+1) = M_i(t) + I_{i,j}(t) \cdot M_i(t)$$

$$R(r_i(t)) = \sum_{n=1}^{N} (w_r)_n \cdot (r_i(t))_n$$

$$A(a_j(t)) = \sum_{m=1}^{M} (w_a)_m \cdot (a_j(t))_m$$



$$F(c_{i,j}(t)) = \frac{1}{\sum_{k=1}^{K} (w_c)_k \cdot (c_{i,j}(t))_k}$$

Migration Model: Attributes

Attractive attributes

- Gross domestic Product (GPD)
- Fragile State Index (FSI)
- Migrants in the country
- Attractive attributes of accessible counties
- Not exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

$$I_{i,j}(t) = R(r_i(t)) \cdot A(a_j(t)) \cdot F(c_{i,j}(t))$$

Migration Model: Attributes

Attractive attributes

- Gross domestic Product (GPD)
- Fragile State Index (FSI)
- Migrants in the country
- Attractive attributes of accessible counties
- Not exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Repulsive attributes

- Gross domestic product (GPD)
- Fragile State Index (FSI)
- Exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

$$I_{i,j}(t) = R(r_i(t)) \cdot A(a_j(t)) \cdot F(c_{i,j}(t))$$

Migration Model: Attributes

Attractive attributes

- Gross domestic Product (GPD)
- Fragile State Index (FSI)
- Migrants in the country
- Attractive attributes of accessible counties
- Not exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Repulsive attributes

- Gross domestic product (GPD)
- Fragile State Index (FSI)
- Exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Separation attributes

- Border security actions

$$I_{i,j}(t) = R(r_i(t)) \cdot A(a_j(t)) \cdot F(c_{i,j}(t))$$

Migration Model: Attributes & Parameters

Attractive attributes

- Gross domestic Product (GPD)
- Fragile State Index (FSI)
- Migrants in the country
- Attractive attributes of accessible counties
- Not exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Repulsive attributes

- Gross domestic product (GPD)
- Fragile State Index (FSI)
- Exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Separation attributes

- Border security actions

$$I_{i,j}(t) = R(r_i(t)) \cdot A(a_j(t)) \cdot F(c_{i,j}(t))$$

$$R(r_i(t)) = \sum_{n=1}^{N} (w_r)_n \cdot (r_i(t))_n$$

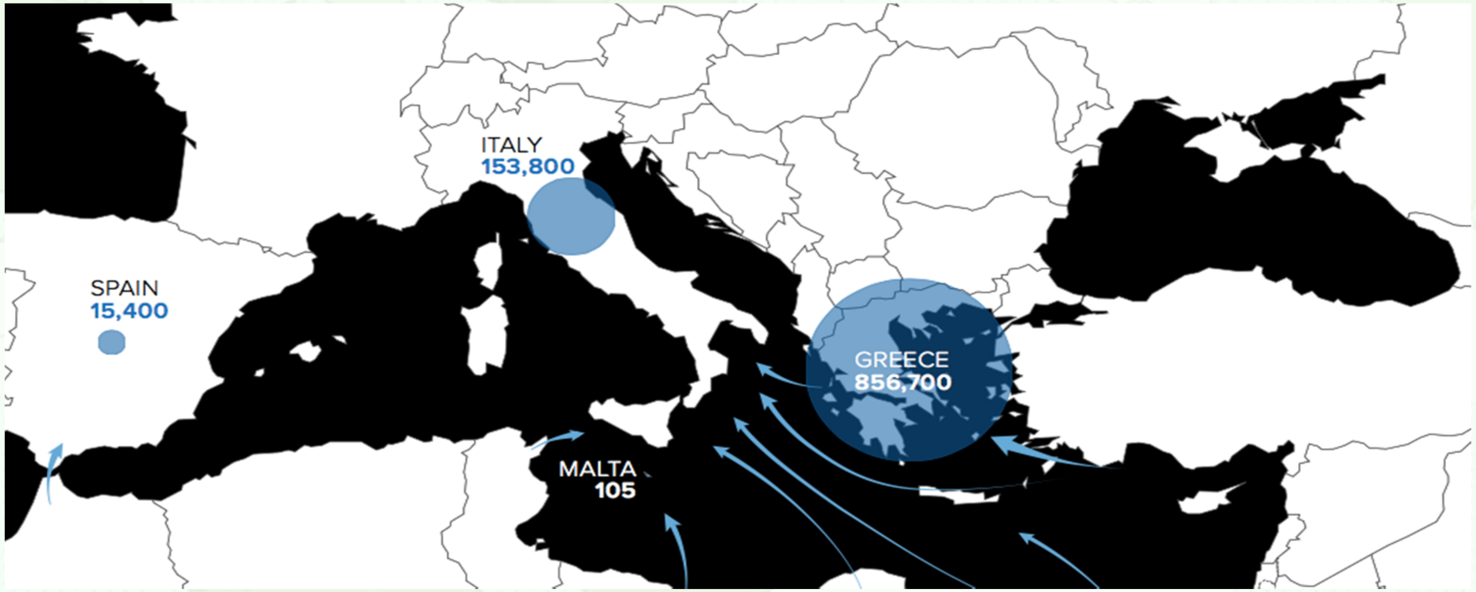
$$A(a_j(t)) = \sum_{m=1}^{M} (w_a)_m \cdot (a_j(t))_m$$

- Parameters

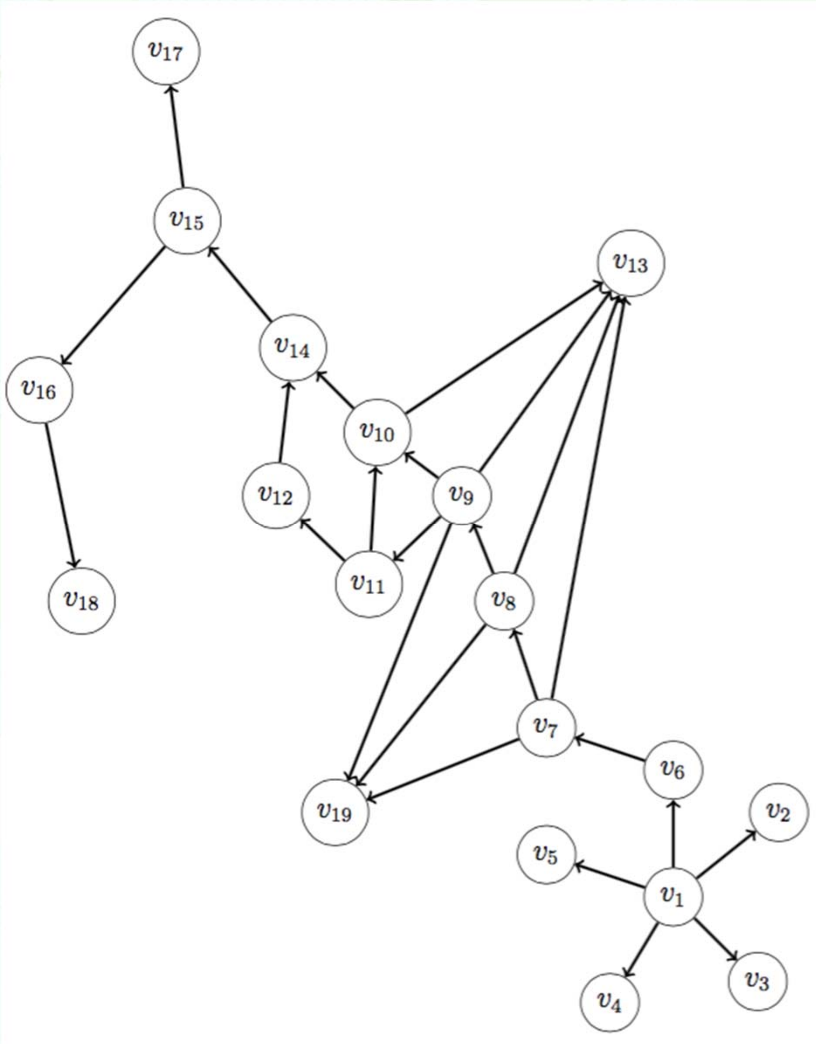
$$F(c_{i,j}(t)) = \frac{1}{\sum_{k=1}^{K} (w_c)_k \cdot (c_{i,j}(t))_k}$$

Refugee Crisis 2015

- Data: Number of asylum applications, partly Transit
- Country of origin: Syria
- Time period: 01.09-31.10.2015

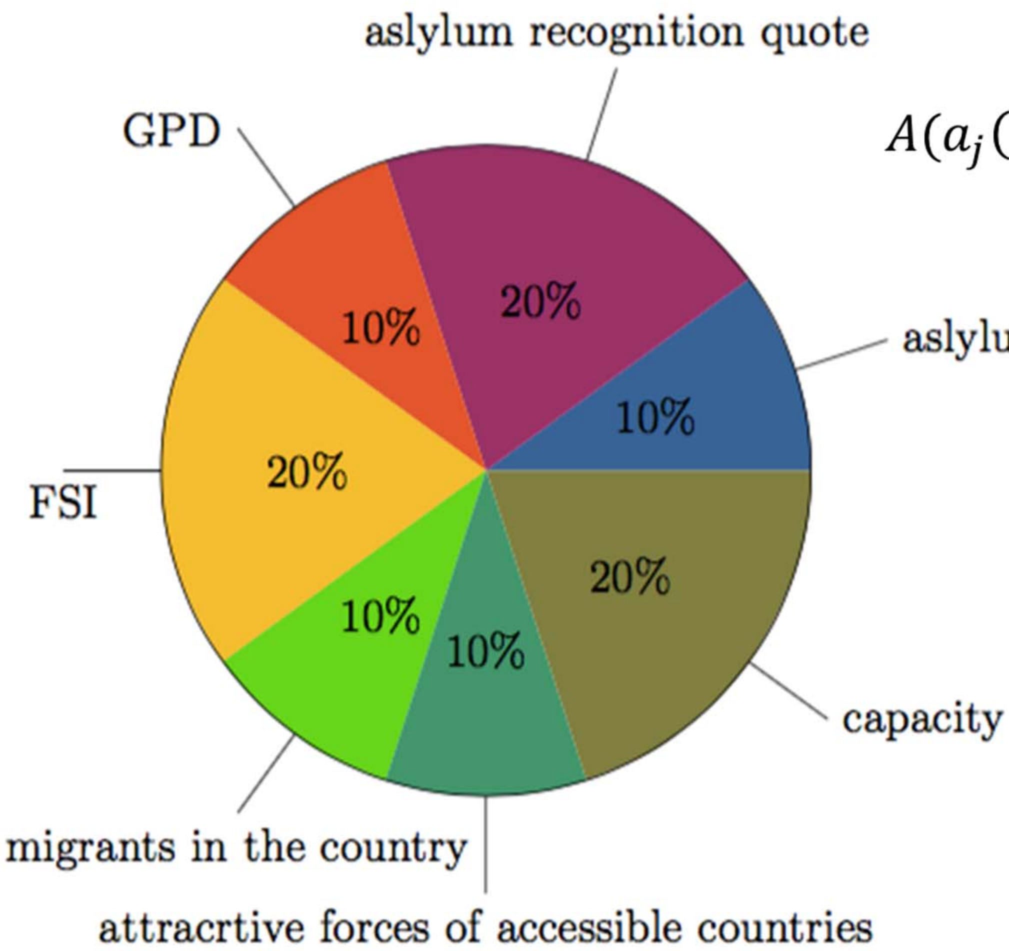


Graph of migration movement



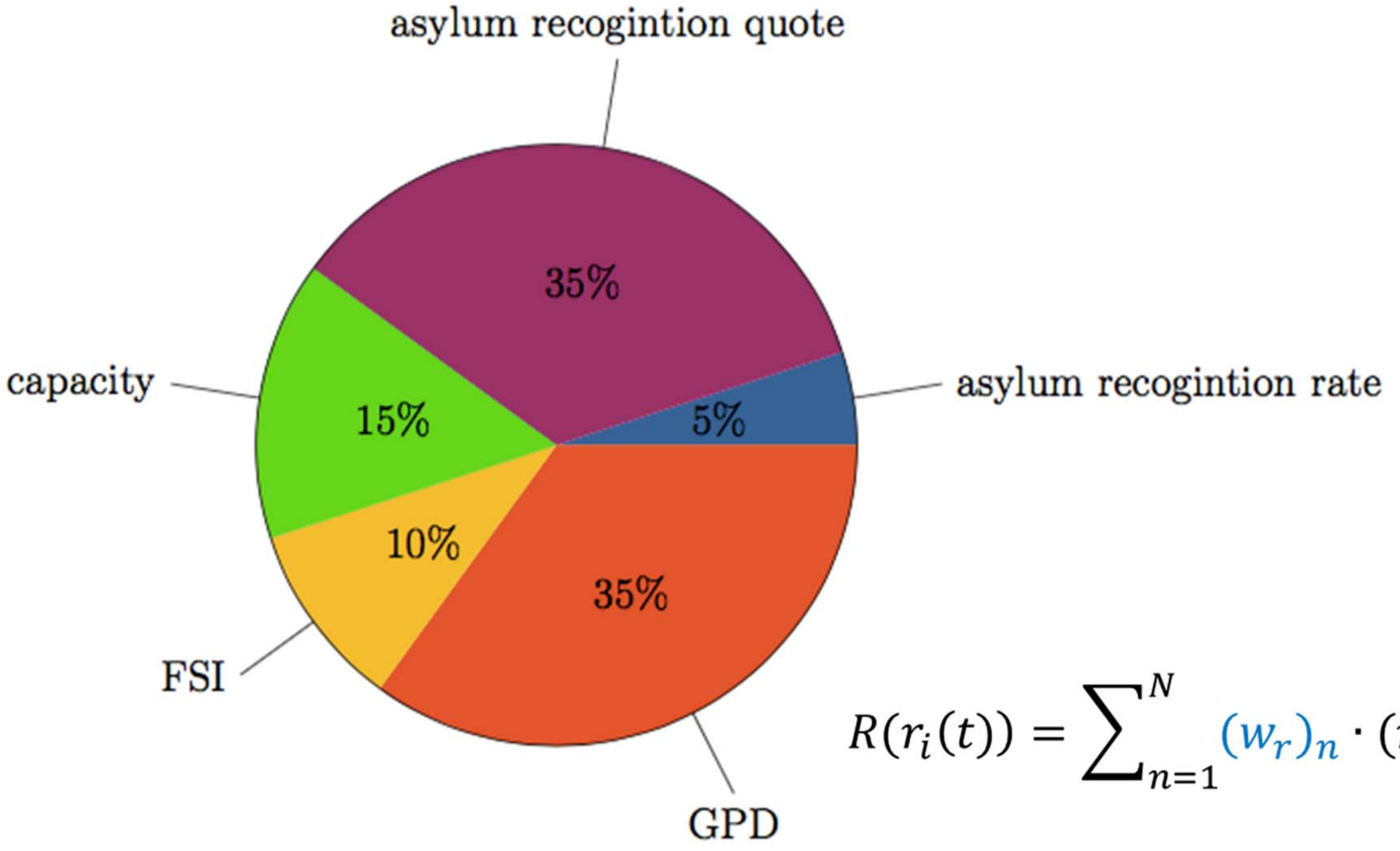
vertices	regions
v_1	Syria
v_2	Iraq
v_3	Jordan
v_4	Egypt
v_5	Lebanon
v_6	Turkey
v_7	Greece
v_8	Republic of Macedonia
v_9	Serbia
v_{10}	Hungary
v_{11}	Croatia
v_{12}	Slovenia
v_{13}	Slovakia, Czech Republic, Rumania, Bulgaria, Poland, Lithuania, Estonia, Latvia
v_{14}	Austria
v_{15}	Germany
v_{16}	UK, Netherlands, Belgium, France
v_{17}	Norway, Finland, Sweden, Denmark
v_{18}	Italy, Spain, Portugal
v_{19}	Albania, Bosnia and Herzegovina, Montenegro

Weighting of attractive attributes: potential destination countries



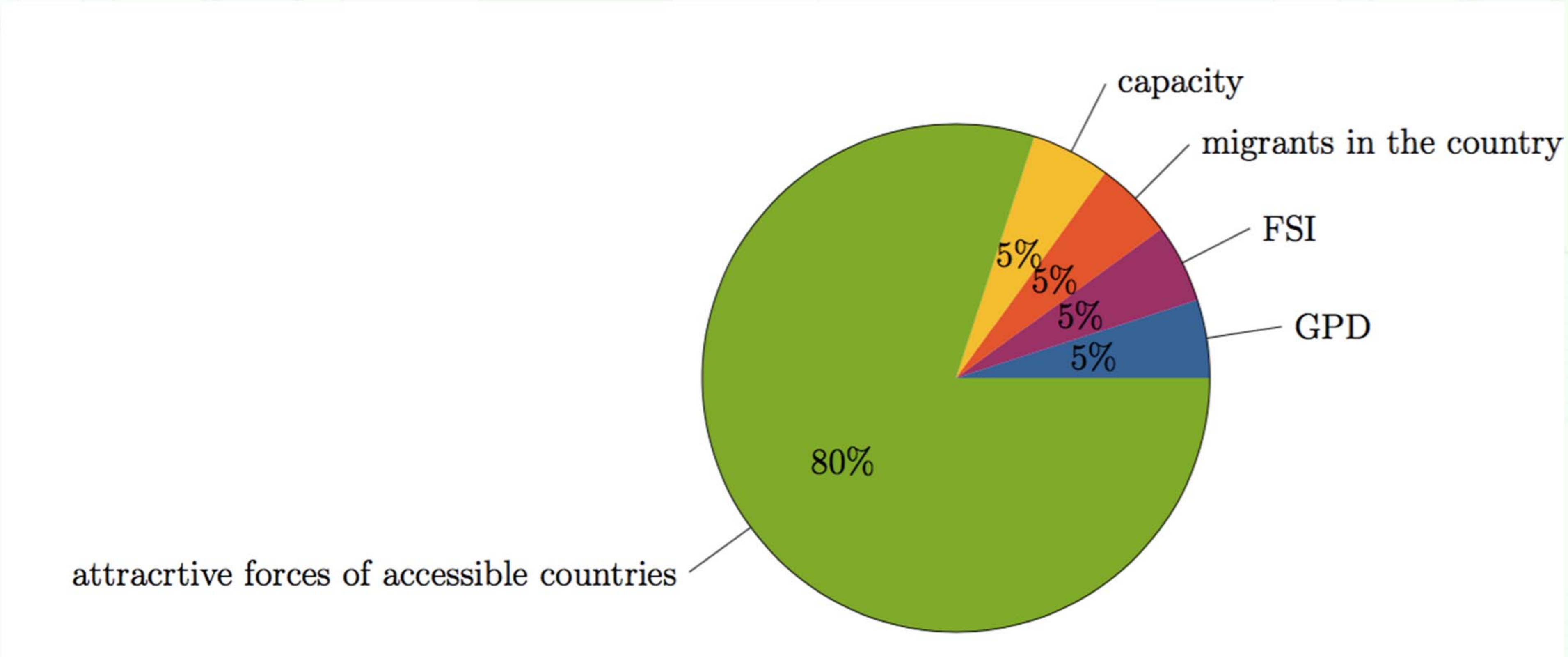
$$A(a_j(t)) = \sum_{m=1}^{M} (w_a)_m \cdot (a_j(t))_m$$

Weighting of the repulsive attributes: potential destination countries



$$R(r_i(t)) = \sum_{n=1}^{N} (w_r)_n \cdot (r_i(t))_n$$

Weighting of the attractive attributes: country of origin



Migration Model: Parameter Identification

Attractive attributes

- Gross domestic Product (GPD)
- Fragile State Index (FSI)
- Migrants in the country
- Attractive attributes of accessible counties
- Not exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Repulsive attributes

- Gross domestic product (GPD)
- Fragile State Index (FSI)
- Exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Separation attributes

- Border security actions

Data: Migration /Day in each region

$$R(r_i(t)) = \sum_{n=1}^{N} (w_r)_n \cdot (r_i(t))_n$$

$$A(a_j(t)) = \sum_{m=1}^{M} (w_a)_m \cdot (a_j(t))_m$$

- Parameters

$$F(c_{i,j}(t)) = \frac{1}{\sum_{k=1}^{K} (w_c)_k \cdot (c_{i,j}(t))_k}$$

Migration Model: Parameter Identification

Attractive attributes

- Gross domestic Product (GPD)
- Fragile State Index (FSI)
- Migrants in the country
- Attractive attributes of accessible counties
- Not exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Repulsive attributes

- Gross domestic product (GPD)
- Fragile State Index (FSI)
- Exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Separation attributes

- Border security actions

Data: Migration /Day in each region

No satisfying Identification

$$R(r_i(t)) = \sum_{n=1}^{N} (w_r)_n \cdot (r_i(t))_n$$

$$A(a_j(t)) = \sum_{m=1}^{M} (w_a)_m \cdot (a_j(t))_m$$

- Parameters

$$F(c_{i,j}(t)) = \frac{1}{\sum_{k=1}^{K} (w_c)_k \cdot (c_{i,j}(t))_k}$$

Migration Model: Parameter Identification

Attractive attributes

- Gross domestic Product (GPD)
- Fragile State Index (FSI)
- Migrants in the country
- Attractive attributes of accessible counties
- Not exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Repulsive attributes

- Gross domestic product (GPD)
- Fragile State Index (FSI)
- Exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Separation attributes

- Border security actions

Data: Migration /Day in each region

No satisfying Identification – only neighbour attraction

$$R(r_i(t)) = \sum_{n=1}^{N} (w_r)_n \cdot (r_i(t))_n$$

$$A(a_j(t)) = \sum_{m=1}^{M} (w_a)_m \cdot (a_j(t))_m$$

- Parameters

$$F(c_{i,j}(t)) = \frac{1}{\sum_{k=1}^{K} (w_c)_k \cdot (c_{i,j}(t))_k}$$

Migration Model: Parameter Identification

Attractive attributes

- Gross domestic Product (GPD)
- Fragile State Index (FSI)
- Migrants in the country
- Attractive attributes of accessible counties
- Not exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Repulsive attributes

- Gross domestic product (GPD)
- Fragile State Index (FSI)
- Exceeded capacity
- Asylum recognition rate
- Asylum recognition quote in Europe

Separation attributes

- Border security actions

Data: Migration /Day in each region

Transit Regions with extended Attraction Attributes

$$R(r_i(t)) = \sum_{n=1}^{N} (w_r)_n \cdot (r_i(t))_n$$

$$A(a_j(t)) = \sum_{m=1}^{M} (w_a)_m \cdot (a_j(t))_m + \sum a_s(t)$$

- Parameters

$$F(c_{i,j}(t)) = \frac{1}{\sum_{k=1}^{K} (w_c)_k \cdot (c_{i,j}(t))_k}$$

Migration Model: Transit Regions

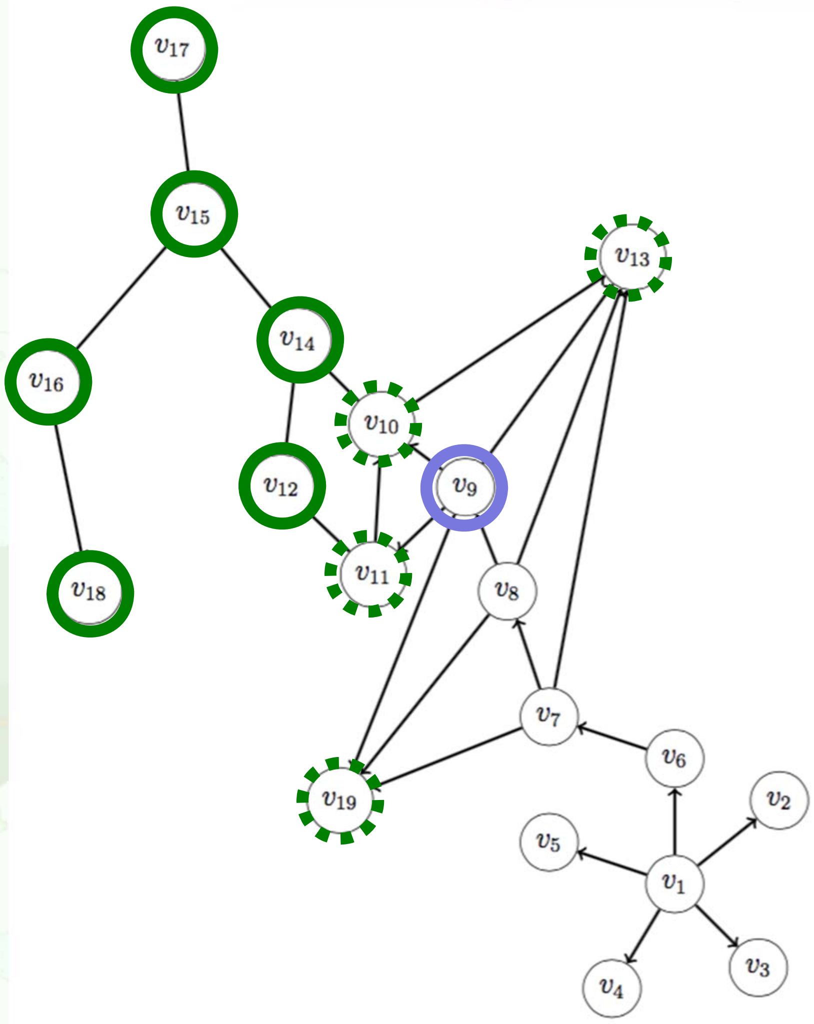
Transit Region

– repulsion $R(r_j(t)) \geq \rho$

– attraction given

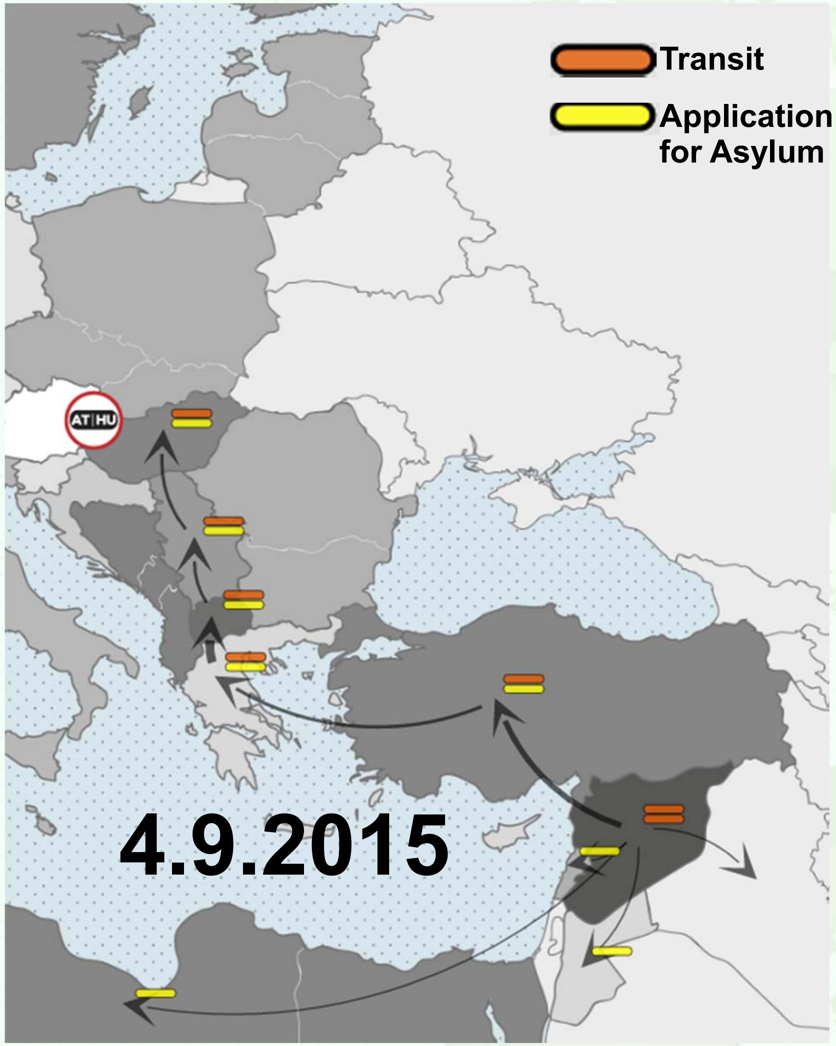
- not only by neighbours,

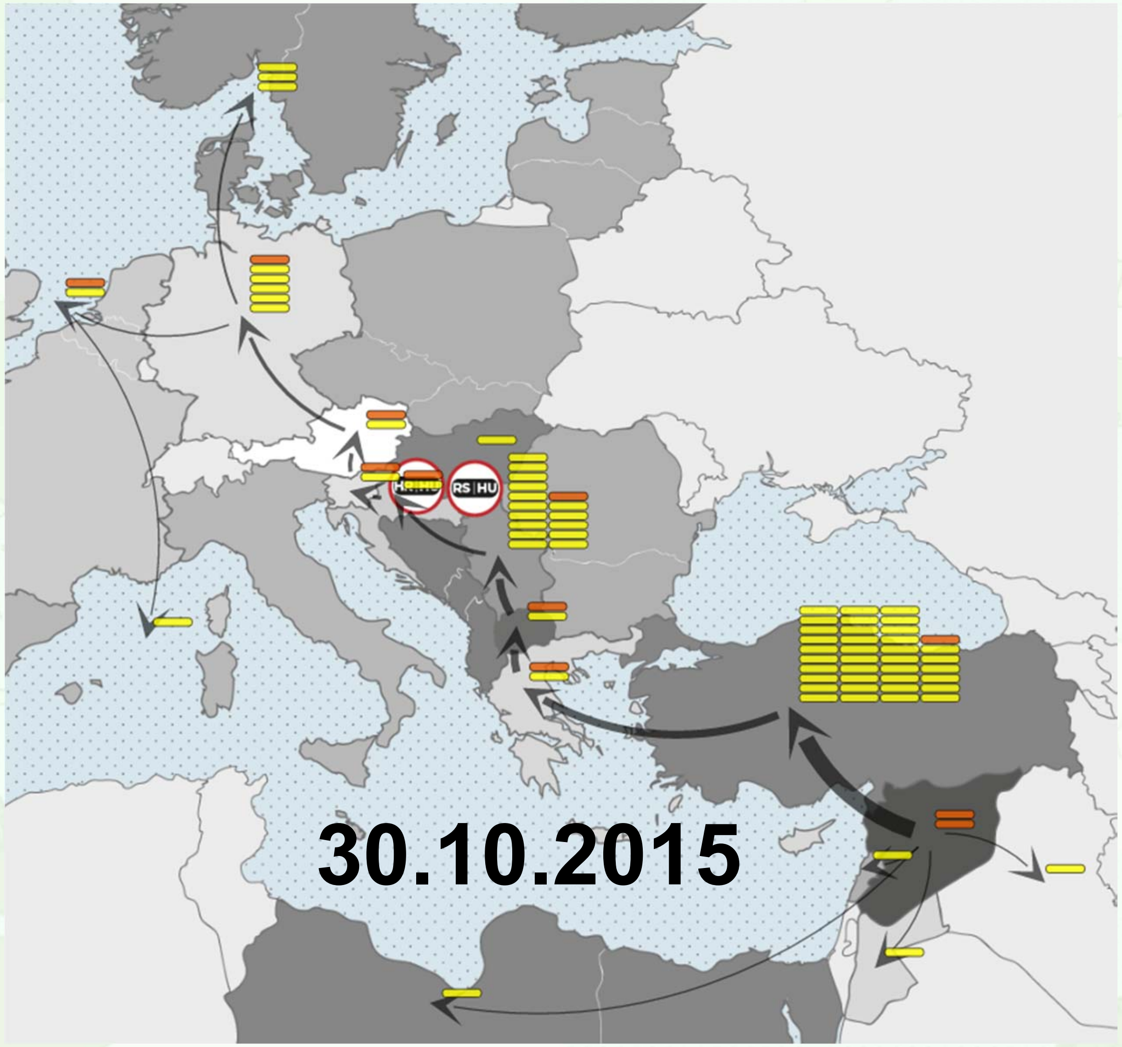
- but also by following regions



$$\tilde{A}(j,t) = \max_{u=j_1,\ldots,j_{\tilde{n}}} \left(A(a_u(t))\right) \cdot \max_{v=j_1,\ldots,j_{\tilde{n}}} F(c_{v,u}(t))$$

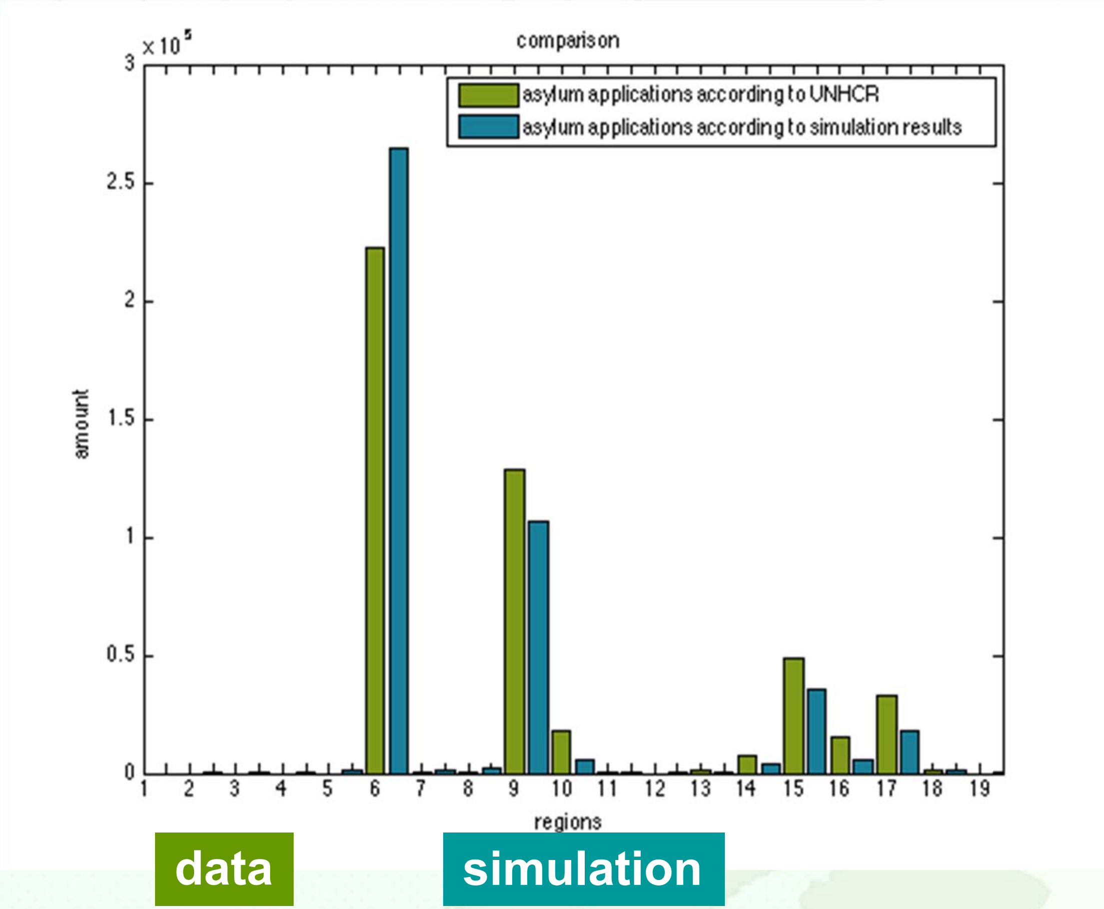
Refugee Crisis 2015 1.9.2015 – 30.10.2015





Modelling + Identification -> Simulation

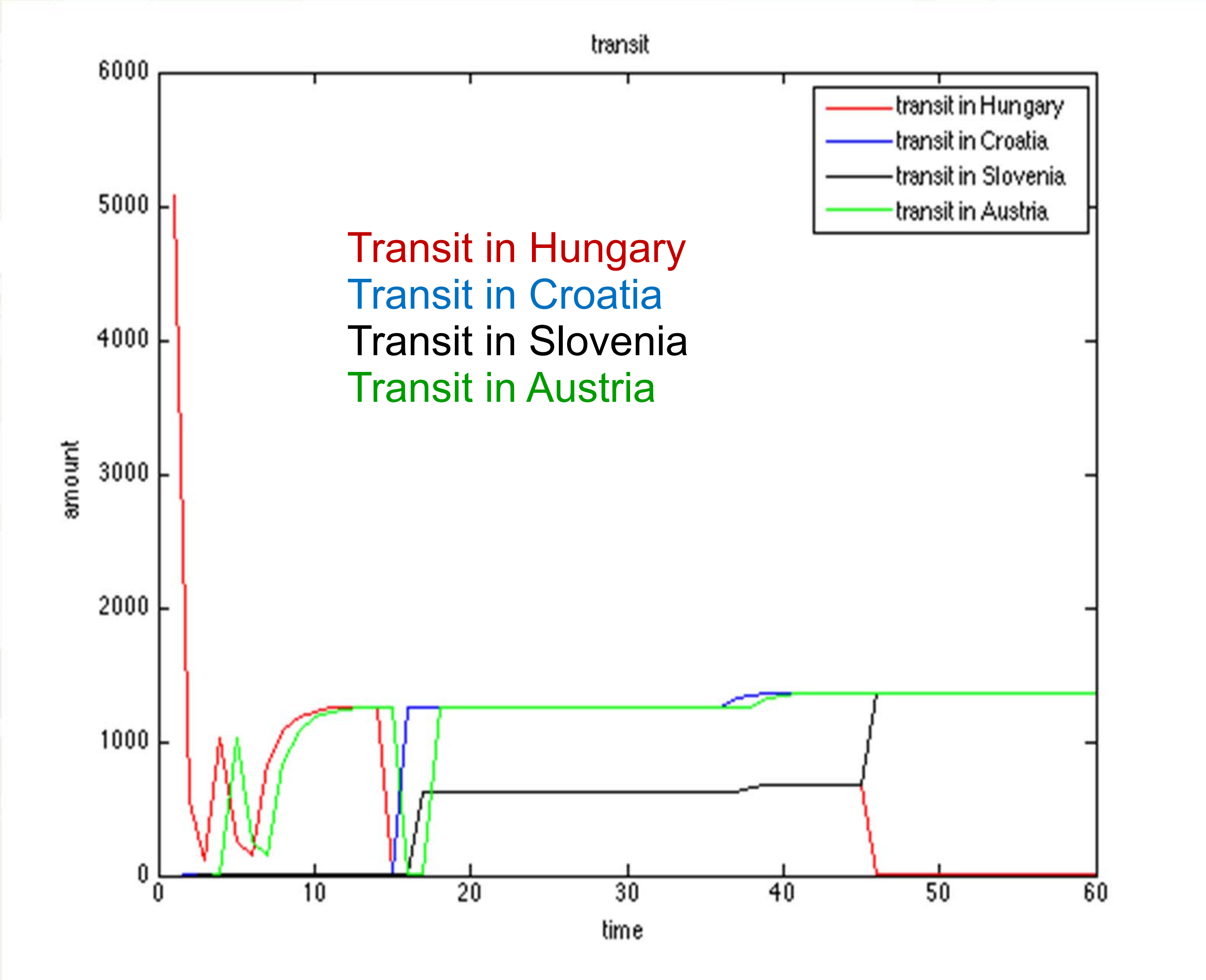
Simulation Results Refugee Crisis: 2015



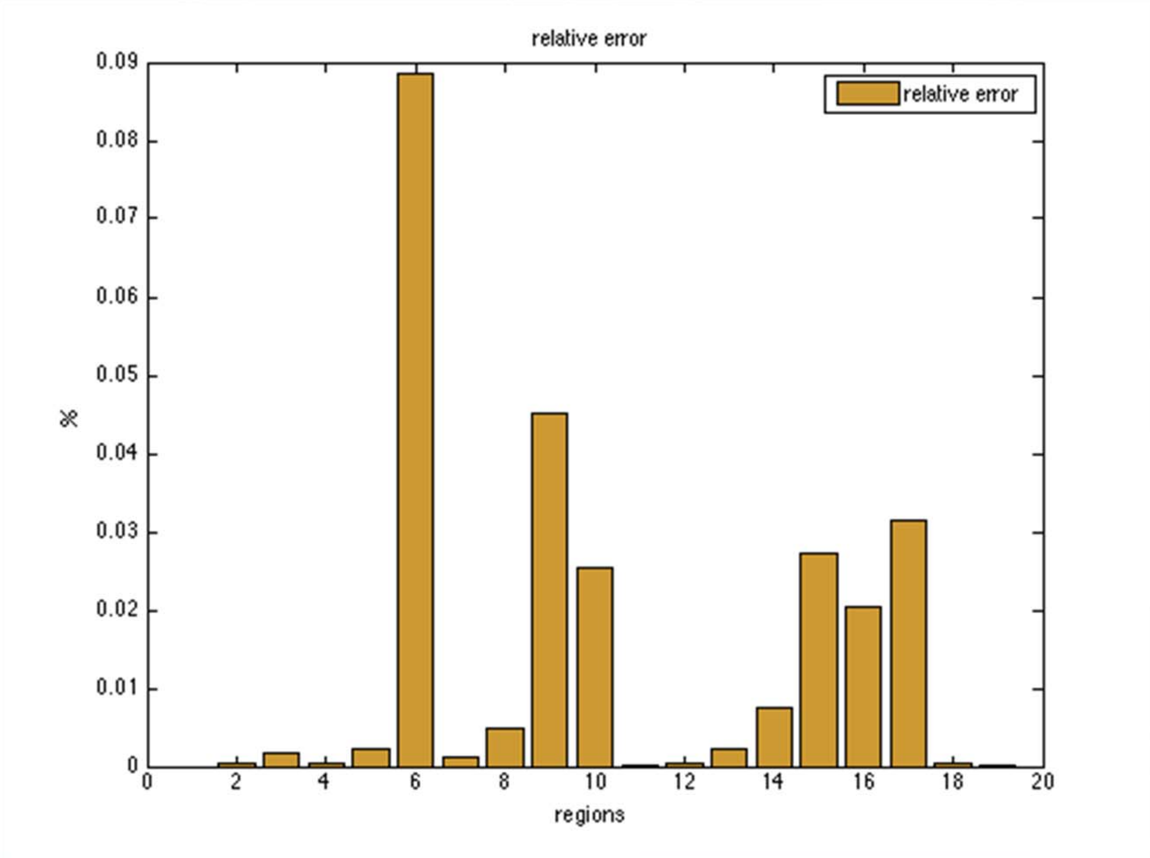
vertices	regions
v_1	Syria
v_2	Iraq
v_3	Jordan
v_4	Egypt
v_5	Lebanon
v_6	Turkey
v_7	Greece
v_8	Republic of Macedonia
v_9	Serbia
v_{10}	Hungary
v_{11}	Croatia
v_{12}	Slovenia
v_{13}	Slovakia, Czech Republic, Rumania, Bulgaria, Poland, Lithuania, Estonia, Latvia
v_{14}	Austria
v_{15}	Germany
v_{16}	UK, Netherlands, Belgium, France
v_{17}	Norway, Finland, Sweden, Denmark
v_{18}	Italy, Spain, Portugal
v_{19}	Albania, Bosnia and Herzegovina, Montenegro

Simulation Results Refugee Crisis 2015: Route Change

- 15.09.2015: alternative route over Croatia
- 15.10.2015: alternative route over Slovenia



Analysis:relative error



vertices	regions
v_1	Syria
v_2	Iraq
v_3	Jordan
v_4	Egypt
v_5	Lebanon
v_6	Turkey
v_7	Greece
v_8	Republic of Macedonia
v_9	Serbia
v_{10}	Hungary
v_{11}	Croatia
v_{12}	Slovenia
v_{13}	Slovakia, Czech Republic, Rumania, Bulgaria, Poland, Lithuania, Estonia, Latvia
v_{14}	Austria
v_{15}	Germany
v_{16}	UK, Netherlands, Belgium, France
v_{17}	Norway, Finland, Sweden, Denmark
v_{18}	Italy, Spain, Portugal
v_{19}	Albania, Bosnia and Herzegovina, Montenegro

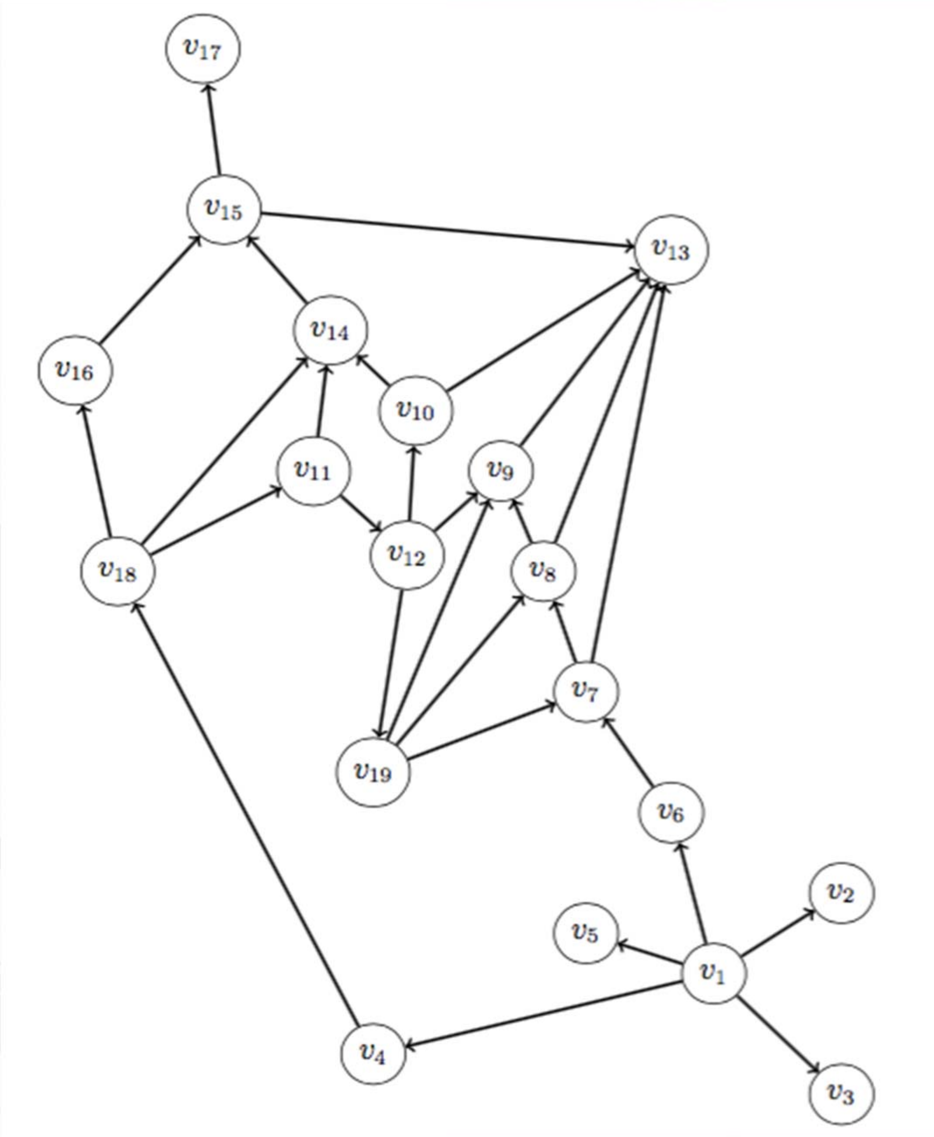
total relative error 25.6 %

Model Characterisation

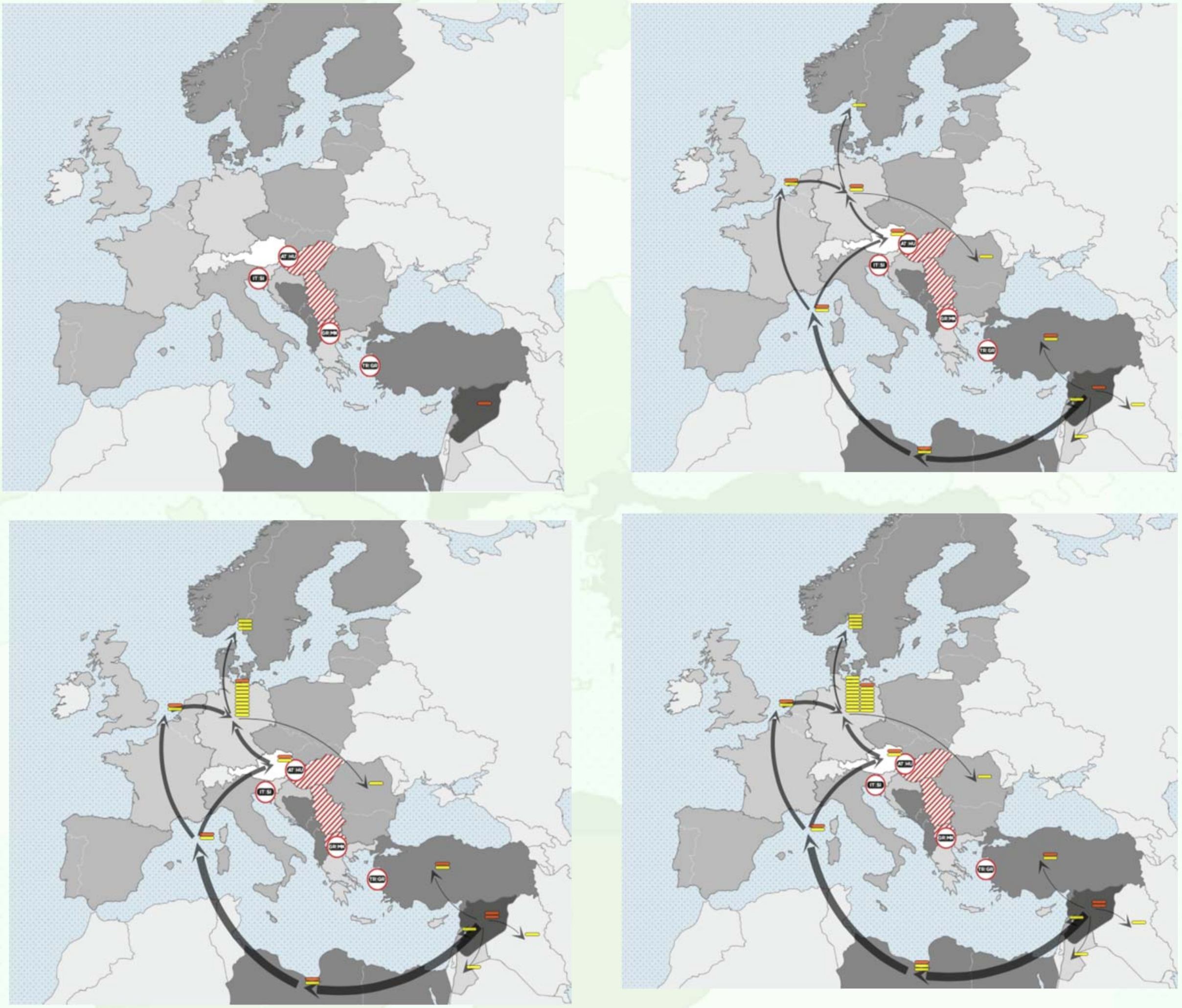
- Macro theoretic model
- Qualitative simulation of migration movement
- Behaviour of populations not individuals
- Model Equations treat static patterns
- Probabilistic model description

Forecast Scenario June 2016

- Time period: June 2016
- Extension of the graph of movement: *central Mediterranean route*
- *Balkan route "closed"*
- *Turkey Deal*



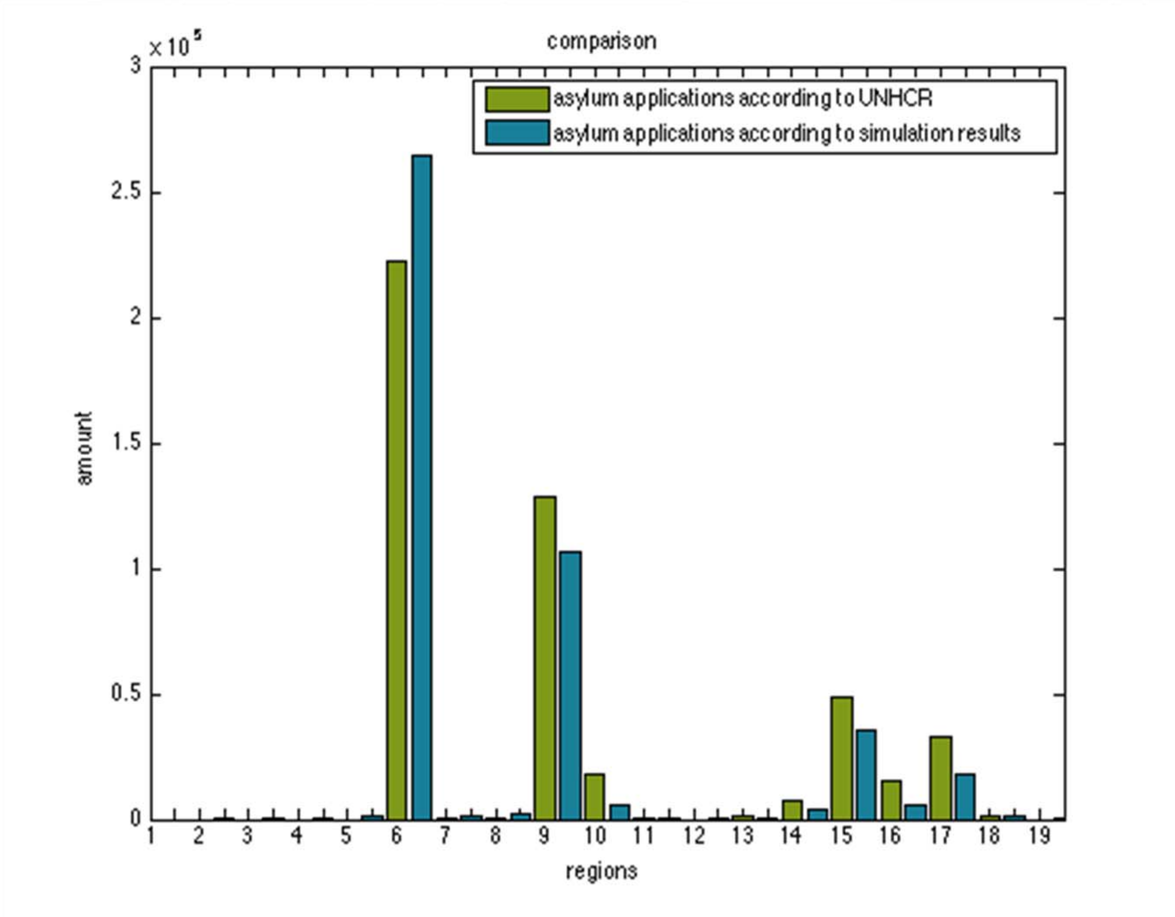
Forecast Scenario June 2016: Visualisation



Visualisation:

Irene Hafner (dwh), Stefan Emrich (dwh), Filip Krasinianski (orf)

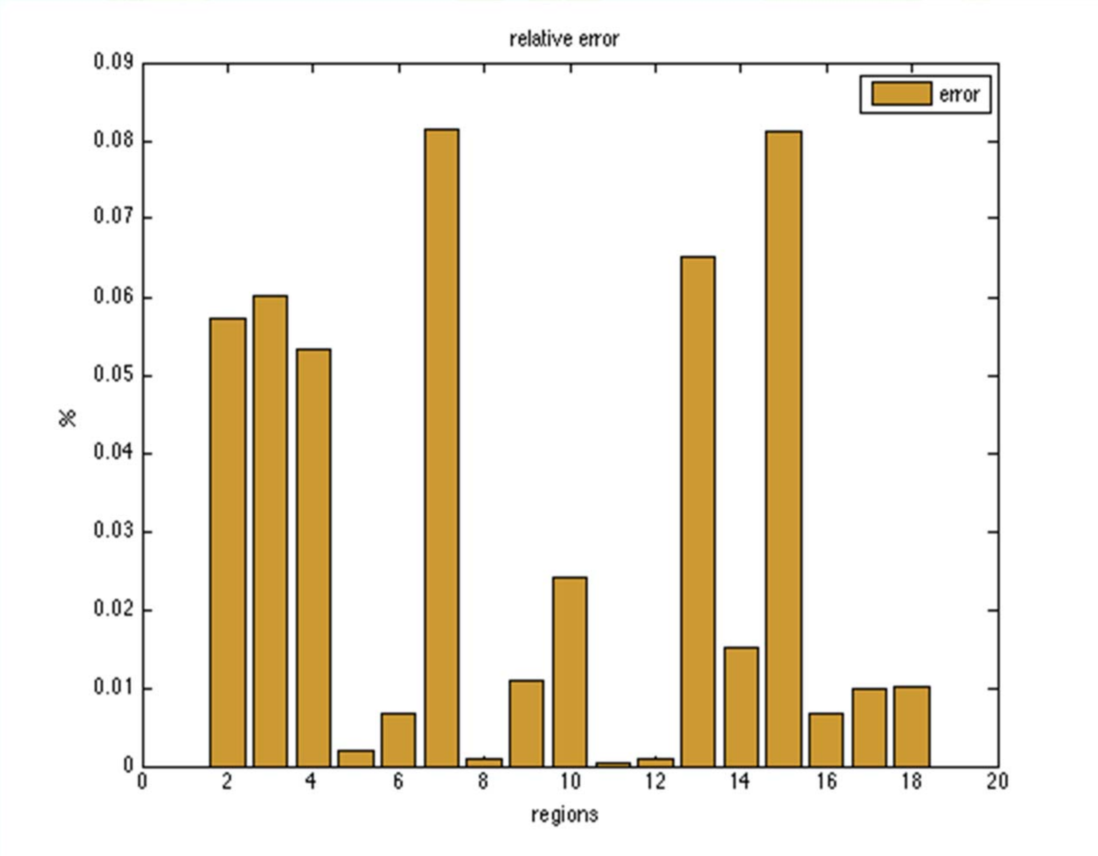
Forecast Scenario June 2016: Simulation Results vs Data (post)



vertices	regions
v_1	Syria
v_2	Iraq
v_3	Jordan
v_4	Egypt
v_5	Lebanon
v_6	Turkey
v_7	Greece
v_8	Republic of Macedonia
v_9	Serbia
v_{10}	Hungary
v_{11}	Croatia
v_{12}	Slovenia
v_{13}	Slovakia, Czech Republic, Rumania, Bulgaria, Poland, Lithuania, Estonia, Latvia
v_{14}	Austria
v_{15}	Germany
v_{16}	UK, Netherlands, Belgium, France
v_{17}	Norway, Finland, Sweden, Denmark
v_{18}	Italy, Spain, Portugal
v_{19}	Albania, Bosnia and Herzegovina, Montenegro

Mathematical Modelling and Simulation

Forecast Scenario June 2016: Error



vertices	regions
v_1	Syria
v_2	Iraq
v_3	Jordan
v_4	Egypt
v_5	Lebanon
v_6	Turkey
v_7	Greece
v_8	Republic of Macedonia
v_9	Serbia
v_{10}	Hungary
v_{11}	Croatia
v_{12}	Slovenia
v_{13}	Slovakia, Czech Republic, Rumania, Bulgaria, Poland, Lithuania, Estonia, Latvia
v_{14}	Austria
v_{15}	Germany
v_{16}	UK, Netherlands, Belgium, France
v_{17}	Norway, Finland, Sweden, Denmark
v_{18}	Italy, Spain, Portugal
v_{19}	Albania, Bosnia and Herzegovina, Montenegro

relative error 8% - sum 48,6 %

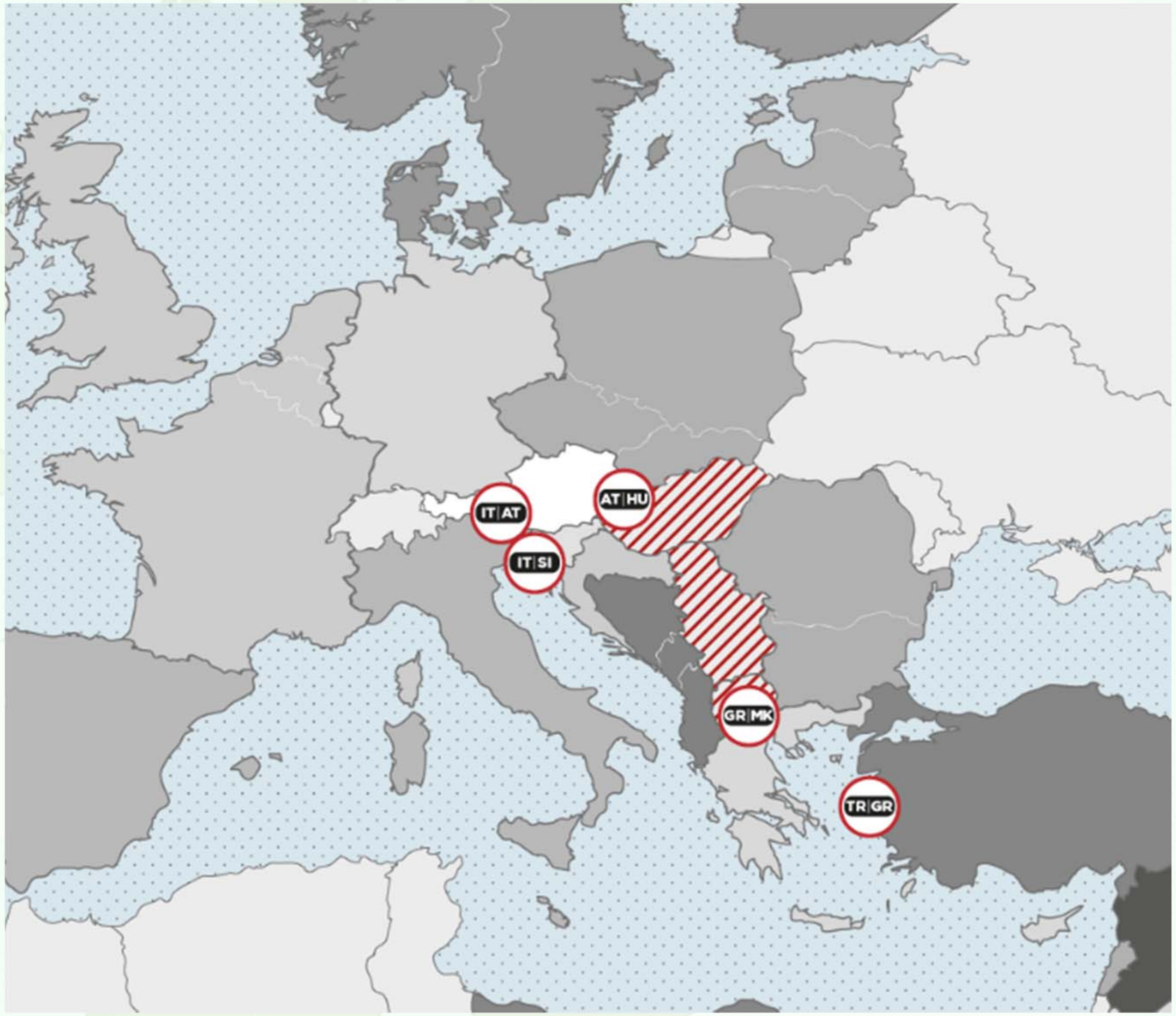
Discussion and outlook

- Qualitative description of migration movement of population groups
- Comparative scenarios can describe all phenomena
- Validity dependent on attributes and weighting
- Include more attribute
- Investigate weighting over longer time period
- Foundation of analysis of influencing factors

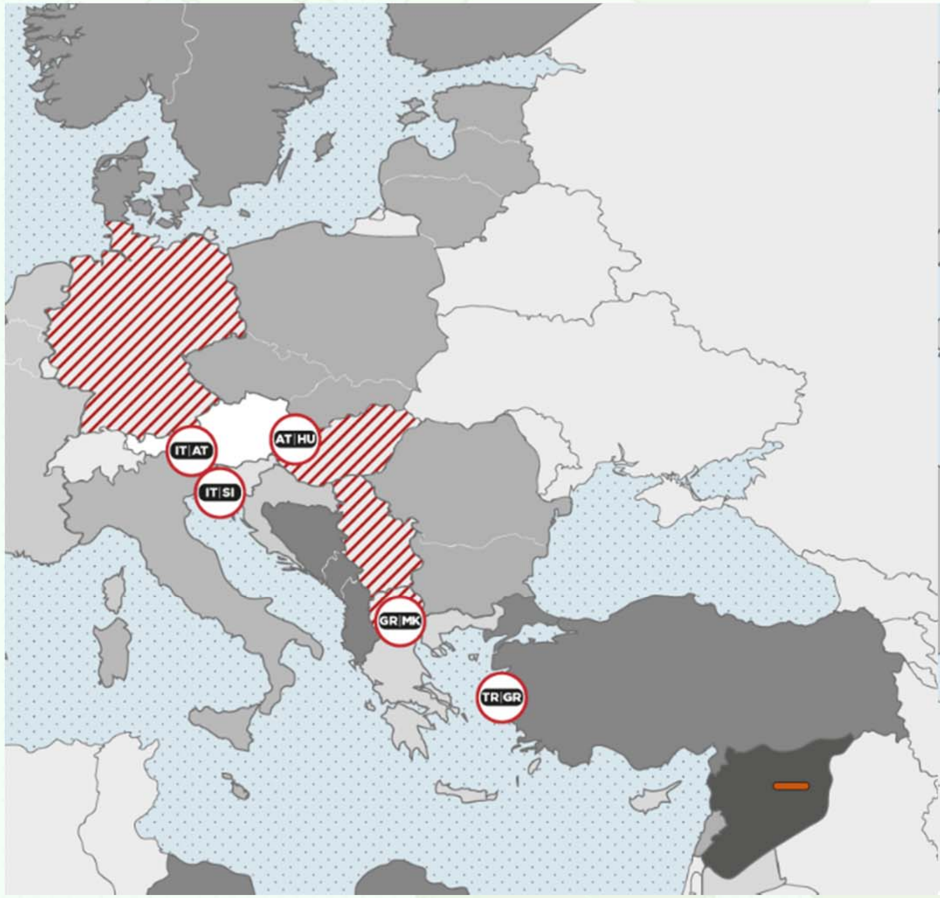
Discussion and outlook

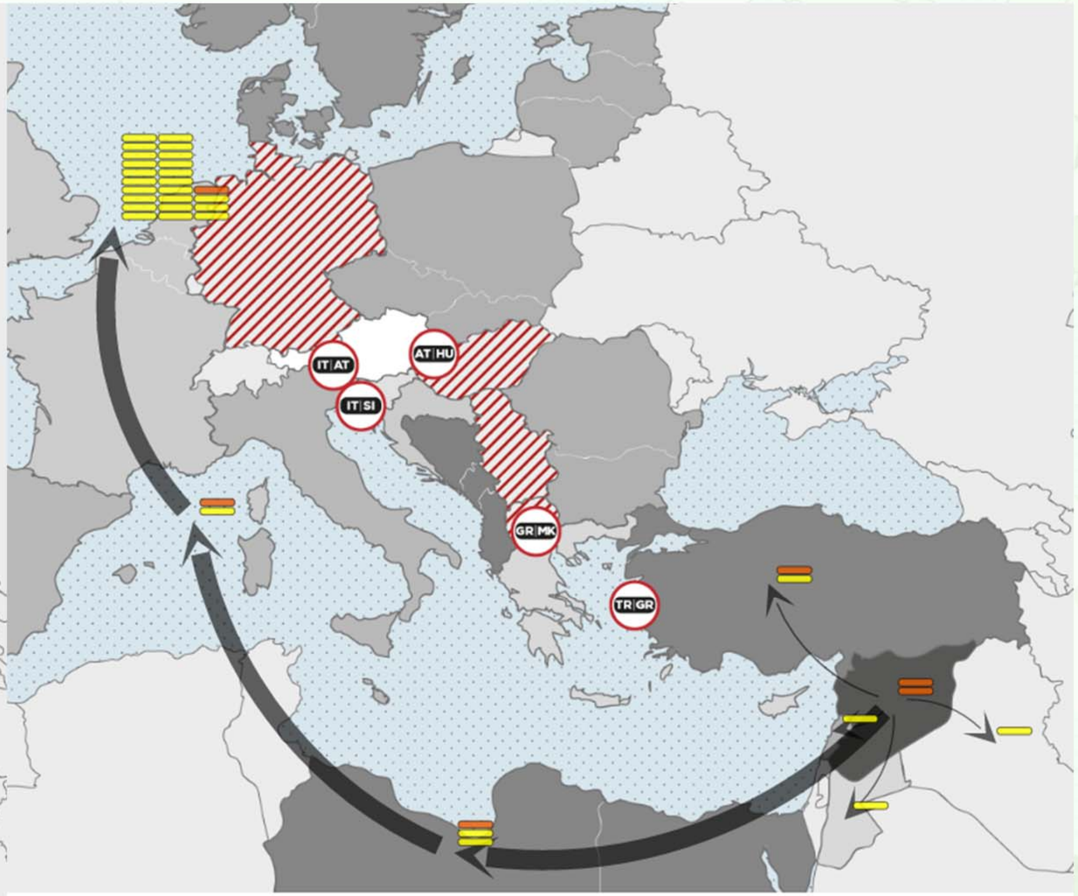
- Qualitative description of migration movement of population groups
- Comparative scenarios can describe all phenomena
- Validity dependent on attributes and weighting
- Include more attribute
- Investigate weighting over longer time period
- Foundation of analysis of influencing factors
- Qualitative Forecast – **What if ……**

What If Brenner Closed - Visualisation

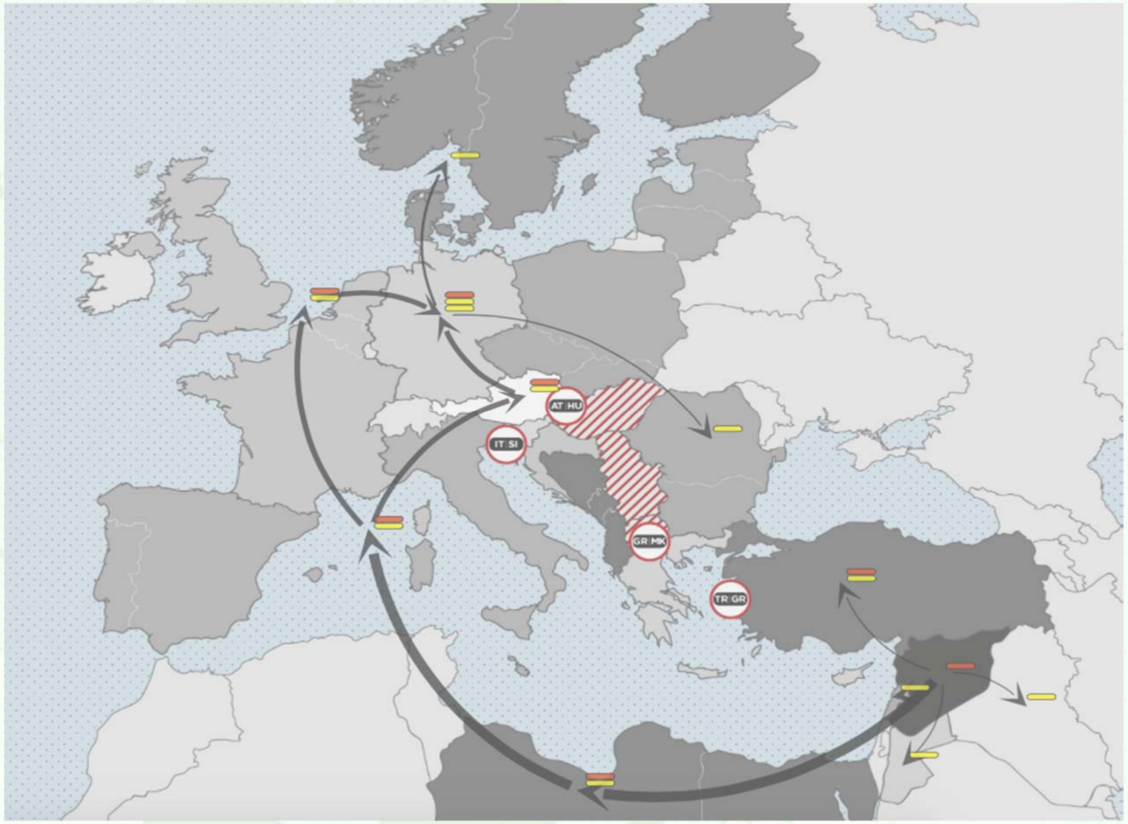


What If Region Closed - Visualisation





Thank you for your attention



Models are in any case a simplification of reality,

- but they should help in better understanding of complex dynamics as migration movement,
- and the intention of this model is to improve the situation of refugees under appropriate prerequisites.