Agent-based modelling using MATLAB

Shan He

School for Computational Science
University of Birmingham

Module 06-23836: Computational Modelling with MATLAB

Outline of Topics

Concepts about Agent-based Modelling

Modelling process of agent-based modelling

A detailed example: BOIDs

Conclusion

What is Agent-based Modelling (ABM)?

- Other names: Agent-based simulation, Individual-based modelling.
- A new modelling method: not equation-based, but based on a population of agents.
- An agent:
  - A discrete entity with its own goals and behaviours
  - Autonomous: with a capability to adapt and modify its beahaviours
- A new research field: have been widely applied to biology, social sciences and other scientific fields and business.

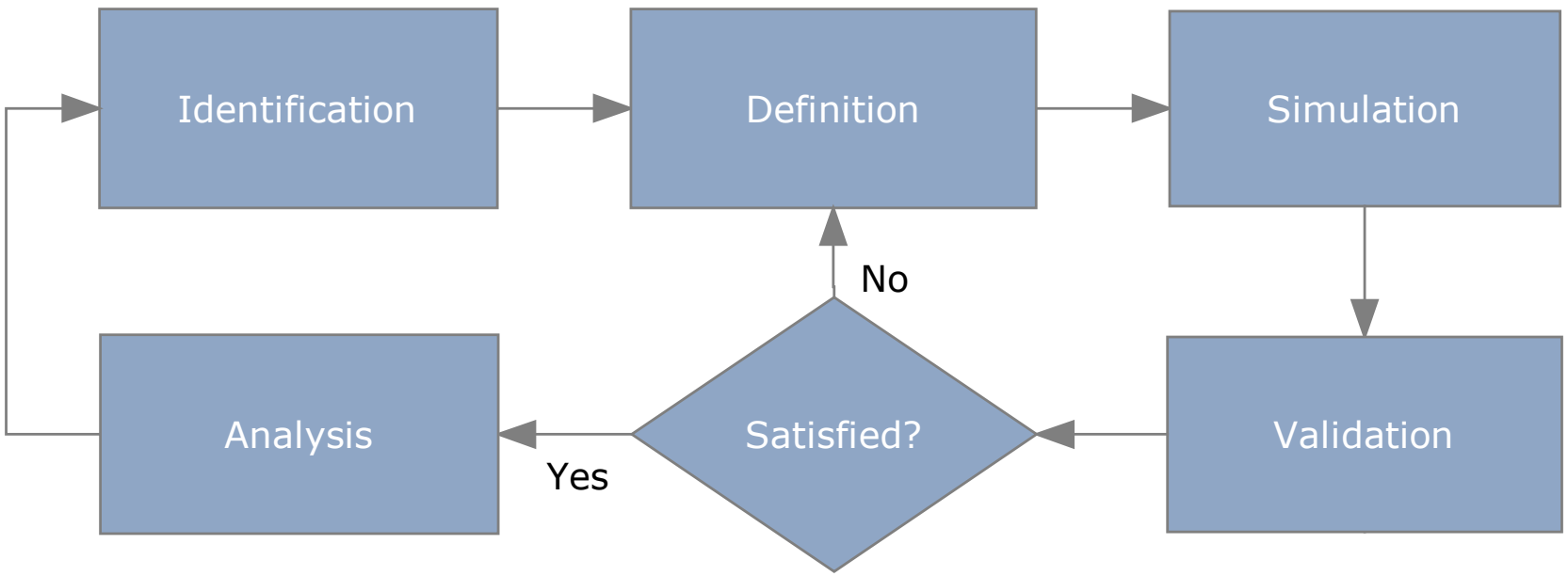
Some examples

- Agent-Based Models in Biology and Medicine
- Agent-based computational economics
- **The Economist**: Agents of change
- **Nature**: Meltdown modelling
- Managing Business Complexity: Discovering Strategic Solutions with Agent-Based Modeling and Simulation

The assumptions behind ABM

- Some key aspect of behaviours can be described.
- Mechanisms by which agents interact can be described.
- Complex behaviours are emerged from simple, usually local interactions.
- The model is built “from the bottom up”.

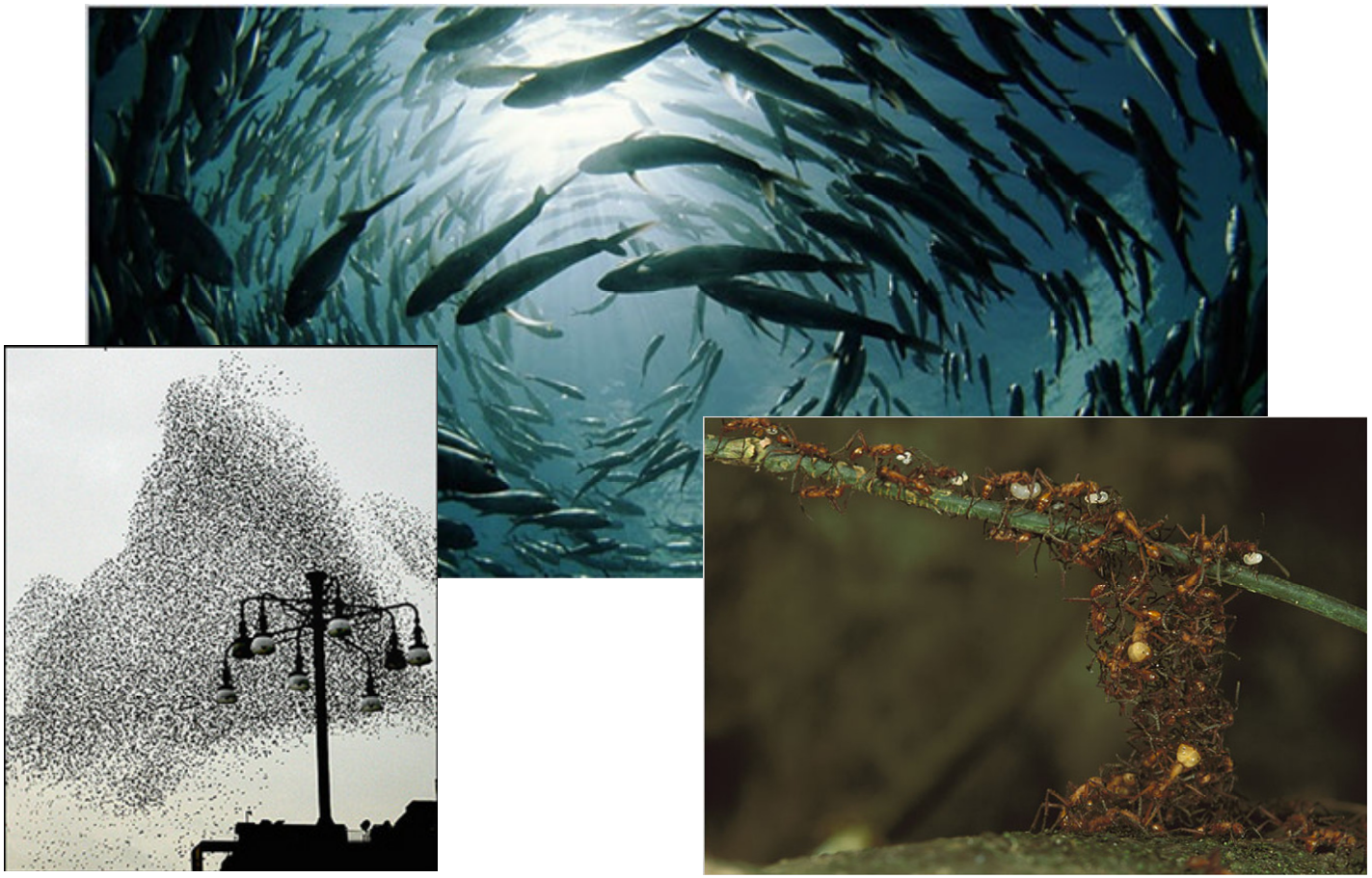
Modelling process



Definition

- **Step 1**: Define the **agent** types and other objects along with their attributes.
- **Step 2**: Define the **environments** the agent will live in and interact with
- **Step 3**: Define **agent method** for updating the agent's attribute in response to their interactions with other agents and the environment.
- **Step 4**: Define **behavioural rules** to specify which agents interact, when they interact and how they interact. This can be changed to other methods, e.g., ANN as we shall see shortly.

Animal swarming behaviour



Pictures from NY Times

Starlings flocking video

BOIDs: ABM of animal schooling/flocking behaviour

- Invented in 1986 by Craig Reynolds to simulate coordinated animal motion such as bird flocks and fish schools.
- A giant step forward in computer animation: animation 'emerges' from complex system
- First film: Stanley and Stella in: Breaking the Ice
- Used in Tim Burton's film *Batman Returns*: bat swarms and "army" of penguins marching through the streets of Gotham City.
- Winner of technical Oscar: 1997 Sci-Tech Awards from Academy of Motion Pictures and Science.

Batman Returns

How BOIDs is developed?

- **Assumption**: the global patterns are the emerging phenomena from interaction of birds or BOIDs.
- Let's follow the definition steps:
- **Step 1**: Identify the agent types and other objects along with their attributes.
- Agent: only Birds, or BOIDs
- Agent attributes: position and velocity

How BOIDs is developed?

- **Step 2**: Define the **environments** the agent will live in and interact with
- A simple 2D or 3D world, no other objects.

How BOIDs is developed?

- **Step 3**: Specify **agent method** for updating the agent's attributes in response to their interactions with other agents and the environment.
- Only consider interactions between BOIDs:- update a BOID's attribute, e.g., position and velocity in response to their interactions with other BOIDs.

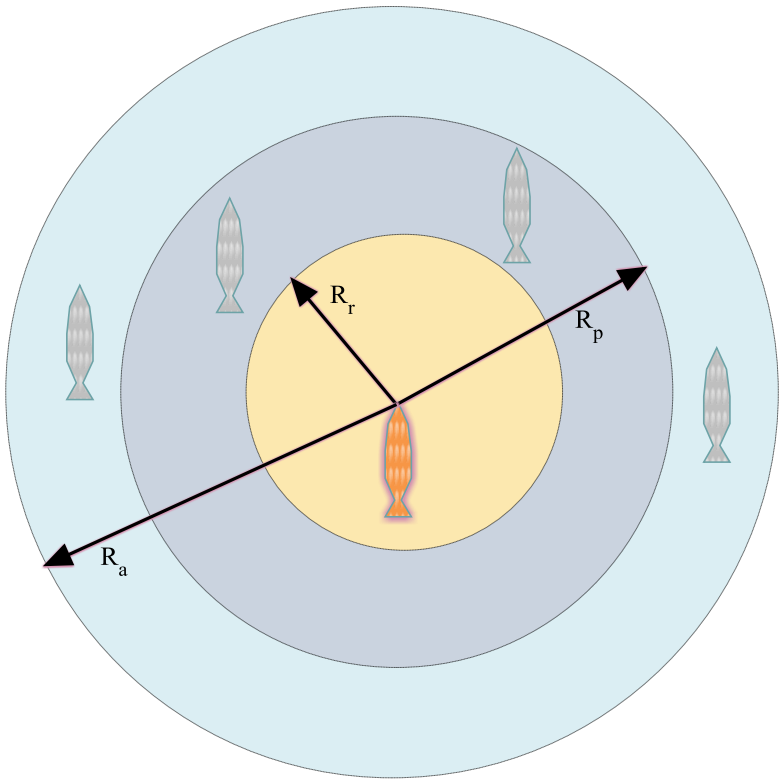
How BOIDs is developed?

- **Step 4**: Define **behavioural rules** to specify which agents interact, when they interact and how they interact.
- The most difficult step: usually done by trail-and-error.
- Sometimes obtestation can be used to derive the behavioural rules and then validate by experiments.
- Ideally, the rules should be derived based on experimental data – data-driven modelling.
- It is possible to automatically generate/evolve behavioural rules by computer, e.g., Artificial Neural Network and Evolutionary Computation.

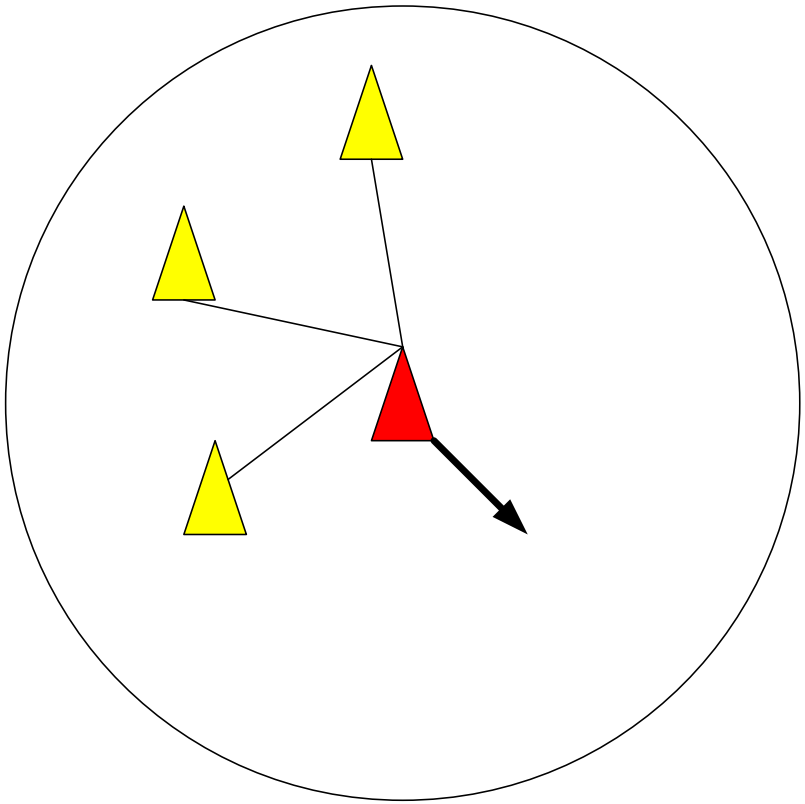
BOIDs: three simple rules

- Repulsion: steer to avoid crowding local flockmates
- Alignment: steer towards the average heading of local flockmates
- Cohesion: steer to move toward the average position of local flockmates

BOIDs: reaction zones

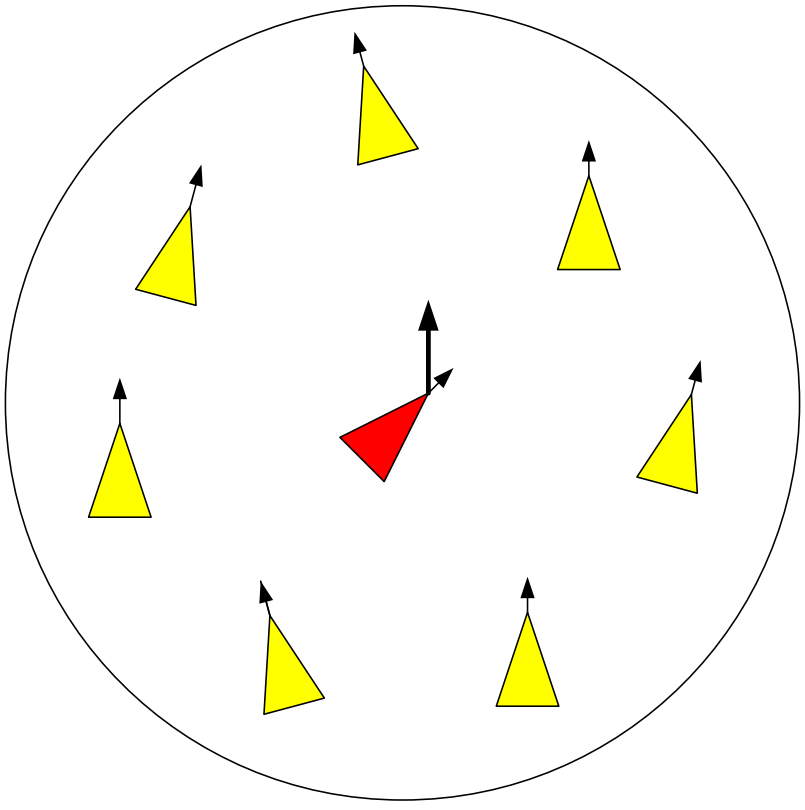


BOIDs: Repulsion



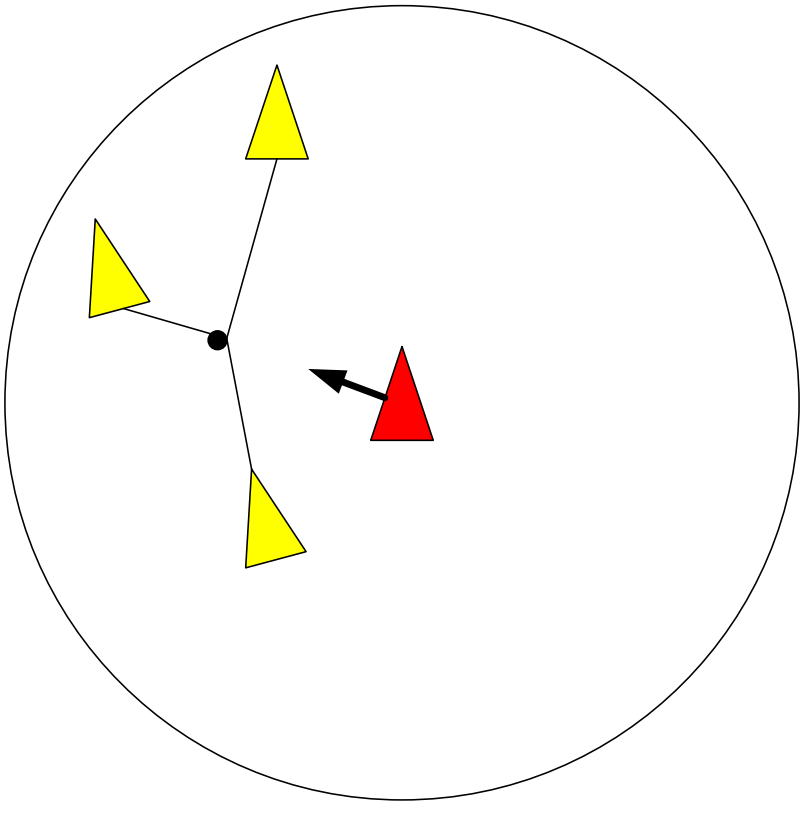
Repulsion zone

BOIDs: Alignment



Alignment zone

BOIDs: Attraction



Attraction zone

Implementing BOIDs in MATLAB

- First question: **What data structures should we used?**
- Requirement: Can easily store/retrive data, ideally should be different types of data, e.g., integer, string or even a data structure.
- **Cell Array** or **Structure Array** or **Object**?
- Cell Array is more convenient for coding, e.g, number as index, but bad for code clarity.
- Object stores data in named fields and also encapsulates the operations perform on data, but comes with complexity for coding.
- Structure Array stores data in named fields and relatively convenient for coding, therefore it is more suitable for our BOID model.

Advantages

- It captures emergent phenomena, therefore, it is a better tool for modelling complex systems.
- It provides a natural description of a system. The behavioural rules can be expressed by natural language, which is easy to be understood.
- It is flexible, e.g., heterogeneous agents and the adaptive/evolving rules.

Disadvantages

- Models too simple: it is very easy to introduce too many assumptions to oversimplified the model.
- Models too complicated: it is also very easy to introduce too many rules to overcomplicate the model.
- Solution: Use the modelling principle and process in the 2nd Lecture.
- Error-prone: many lines of code – very easy to make mistake when coding.
- Difficult to determine model parameters.

Other agent-based modelling tools

- MATLAB is limited in 2D/3D animation.
- Other tools are more powerful:
- Netlogo
- SWARM
- Repast
- MASON
- Processing