Interaction Biases in Multi-Agent Simulations

An Experimental Study

Y. Kubera (yoann.kubera@lifl.fr)
P. Mathieu (philippe.mathieu@lifl.fr)
S. Picault (sebastien.picault@lifl.fr)

University of Lille 1
SMAC Team – LIFL – CNRS-UMR 8022
http://www.lifl.fr/SMAC/

ESAW, *September* 2008

Simulation Design Issues

- Simulation design involves :
  - *domain-specific specialists* that build a model of the simulation;
  - *computer scientists* that implement this model on a particular simulation framework;
- Models may lack information, leading computer scientists to make choices of implementation;
- These choices :
  - may lead to biased results of simulations;
  - are not always made explicitely by the computer scientist.

Towards a non ambiguous domain independent framework

Our Goal

To provide a generic and domain independent simulation methodology and framework.

This requires :

- the identification of all functionnal units underlying the architecture of any simulation;
- the identification of implementation choices for each unit;
- A fine setting of these implementation choices as explicit parameters of the architecture.

Focus of this presentation

Subject of this paper

Study a particular parameter that specifies *"in which actions or interactions an agent may participate in simultaneously ?"*

Without a precise specification of this point, implementation is likely to be biased.

Outline

What are Interactions ?

All actions in a simulation use the same overall pattern :

- They are performed by an agent (the *Source*);
- They are triggered only if some conditions are met;
- If conditions are met, the source acts.

Interaction

An interaction is an action that involves another agent than the *Source* (Reproduce, Hunt, Pick Up, . . .).
This other agent is called *Target*

Decomposition Overview

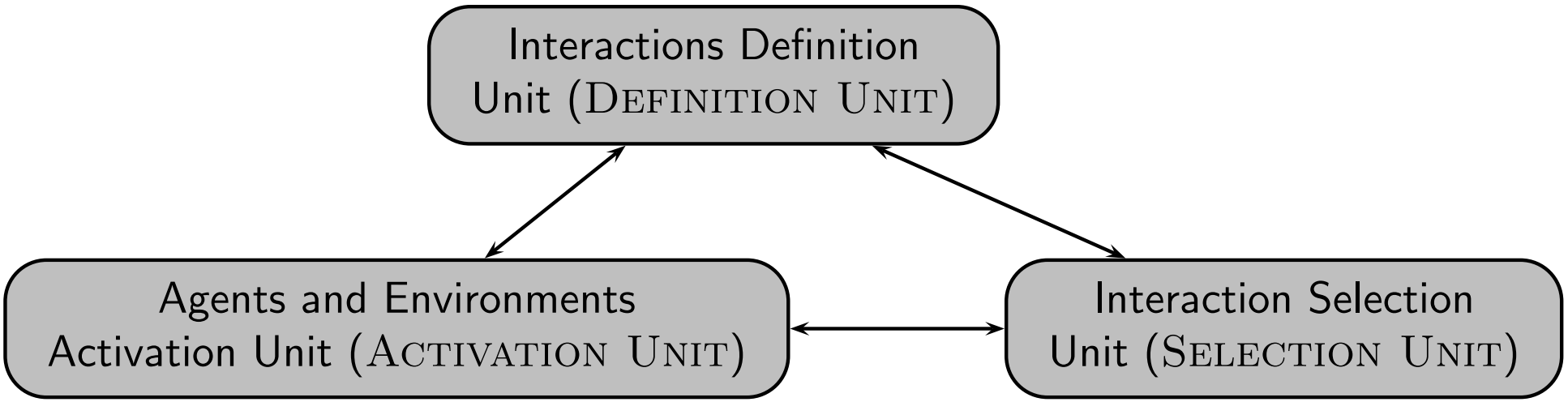


Figure: The three main functional units of a multi-agent simulation.

Decomposition Overview

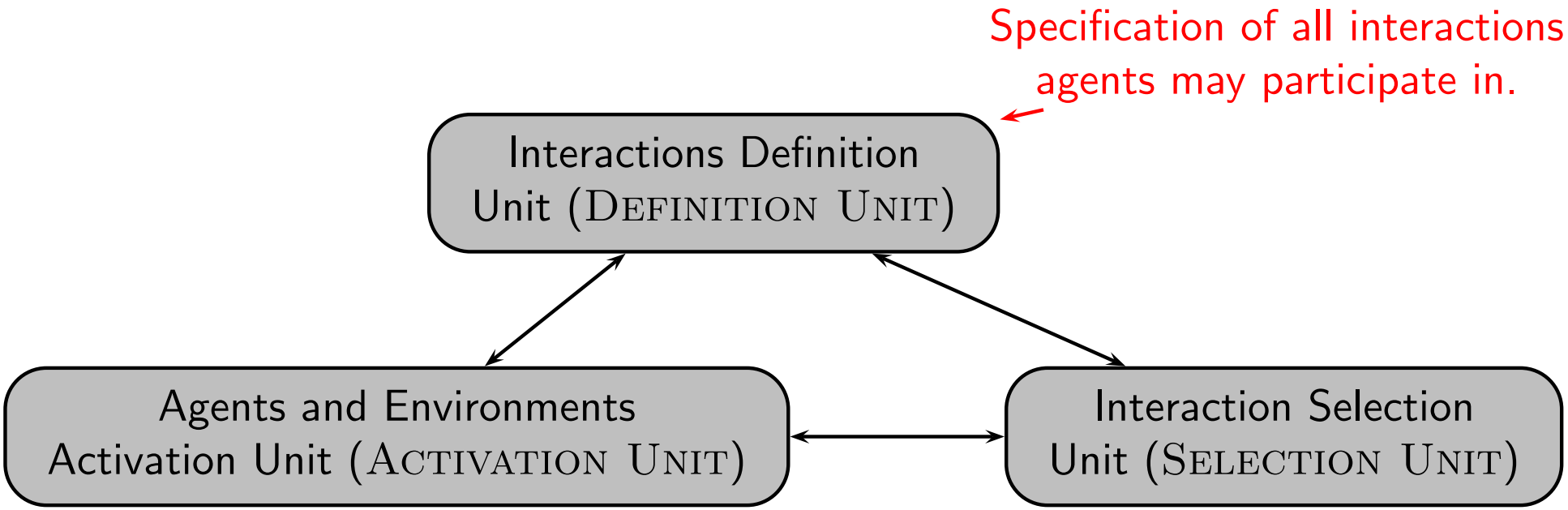


Figure: The three main functional units of a multi-agent simulation.

Decomposition Overview

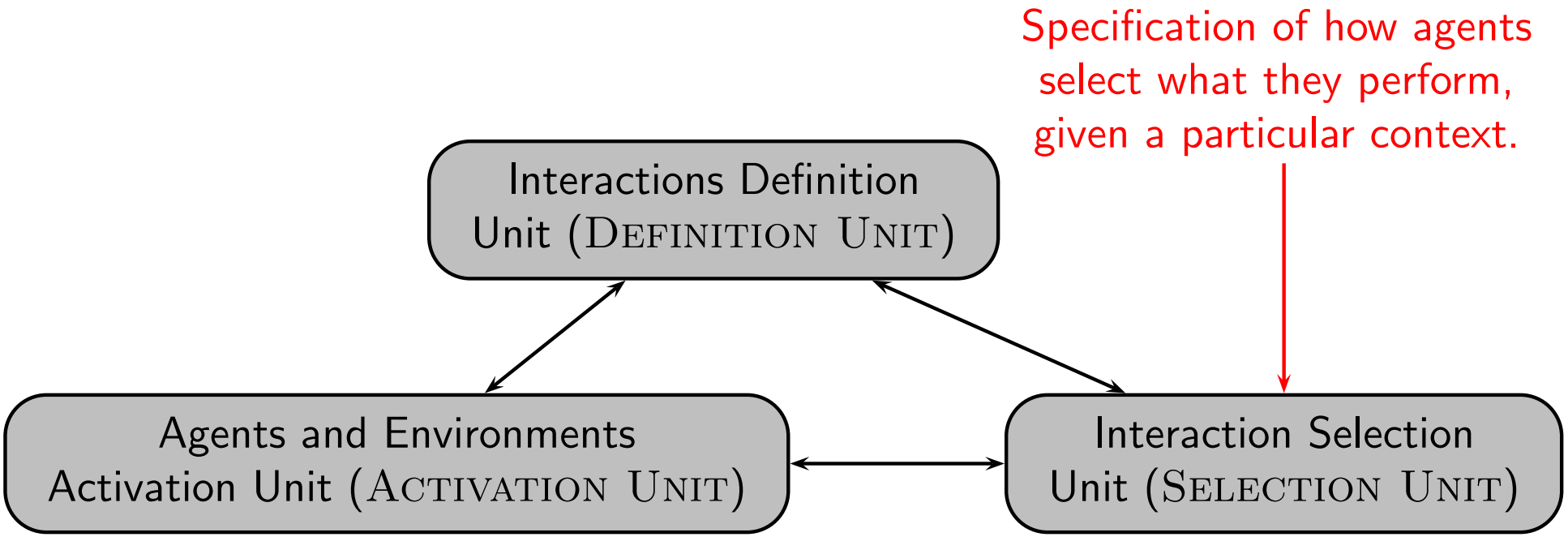


Figure: The three main functional units of a multi-agent simulation.

Decomposition Overview

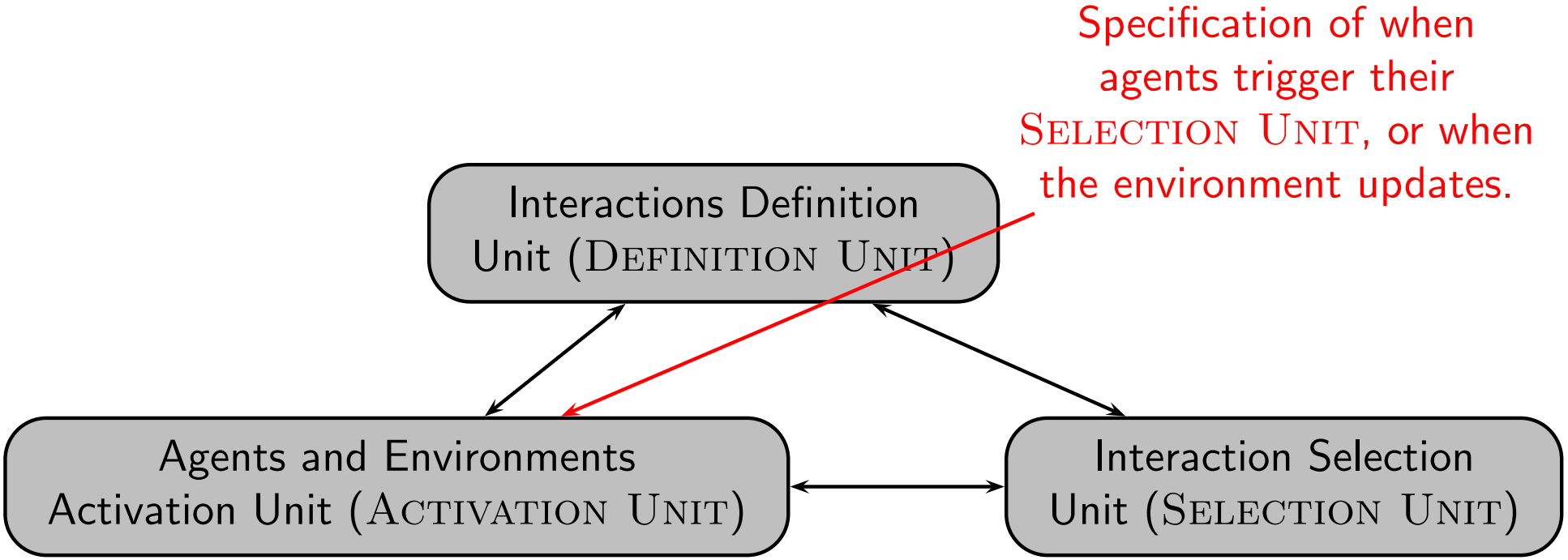


Figure: The three main functional units of a multi-agent simulation.

Simulation's Execution

A simulation is a repetition of 3-steps sequences :

1. The Activation Unit either :
   - selects the next agent that will behave, and goes to step 2;
   - updates the environment, and does step 1 again;
2. The Activation Unit builds agent's perceived affordances thanks to the informations in the Definition Unit;
3. The Selection Unit selects one of those affordances with a particular selection policy, and executes it.

Simulation's Execution

All simulations use implicitly this decomposition.

For instance, in a simulation made with Netlogo where agents reproduce and wander :

```
to go
  ask turtles [go-turtle]
end

to go-turtle
  ifelse any? other turtles-here [
    hatch 1 [ fd 1 ]
  ][
    right 90
    forward 2
  ]
 end
```

Simulation's Execution

All simulations use implicitly this decomposition.

For instance, in a simulation made with Netlogo where agents reproduce and wander :

Activation Unit

```
to go
    ask turtles [go-turtle]
end

to go-turtle
  ifelse any? other turtles-here [
    hatch 1 [ fd 1 ]
  ][
    right 90
    forward 2
  ]
 end
```

Simulation's Execution

All simulations use implicitly this decomposition.

For instance, in a simulation made with Netlogo where agents reproduce and wander :

Activation Unit

Definition Unit:

- Reproduction interaction
- Wander action

```
to go
   ask turtles [go-turtle]
end

to go-turtle
  ifelse any? other turtles-here [
     hatch 1 [ fd 1 ]
  ][
     right 90
     forward 2
  ]
 end
```

Simulation's Execution

All simulations use implicitly this decomposition.

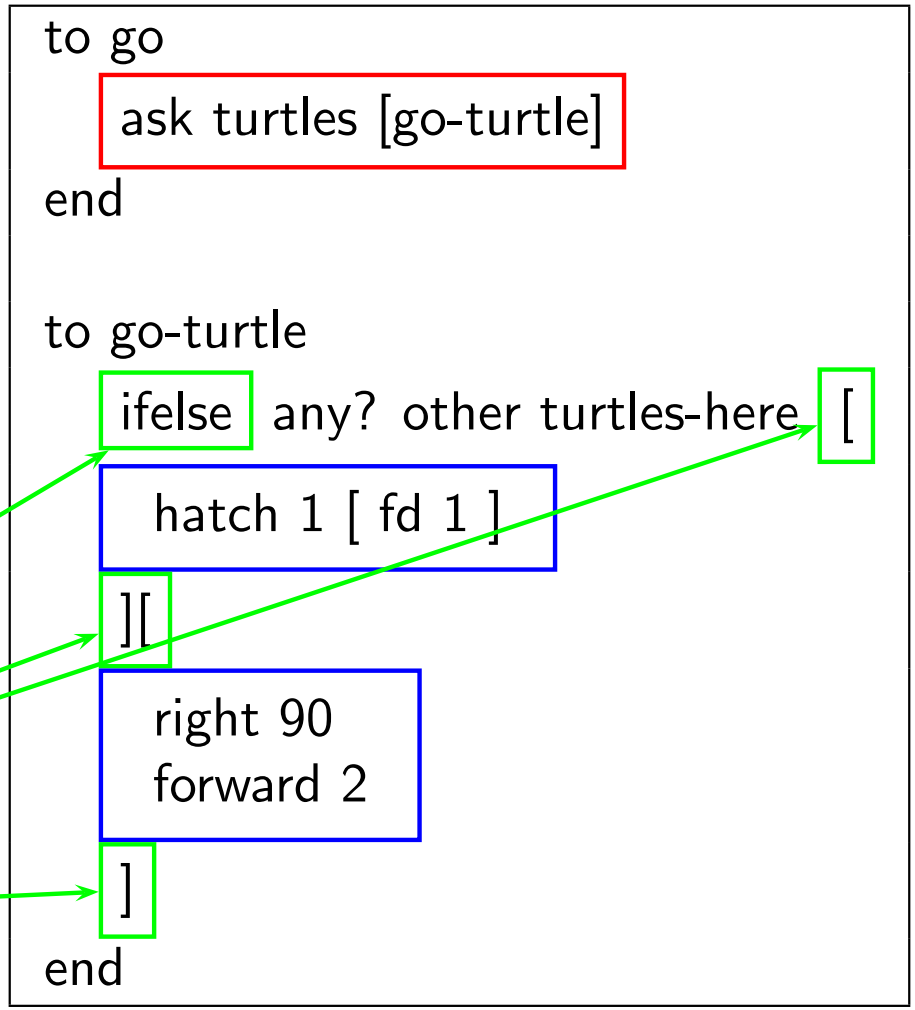
For instance, in a simulation made with Netlogo where agents reproduce and wander :

Activation Unit

Definition Unit:

- Reproduction interaction
- Wander action

Selection Unit



Focus of the Study : the Activation Unit

A simulation is a repetition of 3-steps sequences :

1. The Activation Unit either :
   - *selects the next agent that will behave*, and goes to step 2;
   - updates the environment, and does step 1 again;
2. The Activation Unit *builds agent's perceived affordances* thanks to the informations in the Definition Unit;
3. The Selection Unit selects one of those affordances with a particular selection policy, and executes it.

Outline

Bias Example (1/3)

The Model

- A **Food** agent :
  - has an attribute *quantity*;
- A **Eater** agent :
  - has an attribute *energy*;
  - *reproduces* with another close **Eater** agent;
  - or *eats* a particular quantity of a nearby **Food** agent;
  - or *wanders* in the environment;

Bias Example (2/3)

Time model in this example

- Time is discrete (simulation executes by time steps $t \in \mathbb{N}$);
- Time is asynchronous (at a time t, every agent acts one after the other in an order $\mathbb{O}_t$);

Expected Behavior

- An **Eater** may reproduce only once at a time;
- Many **Eater** may *eat* the same food at the same time.

Bias Example (3/3)

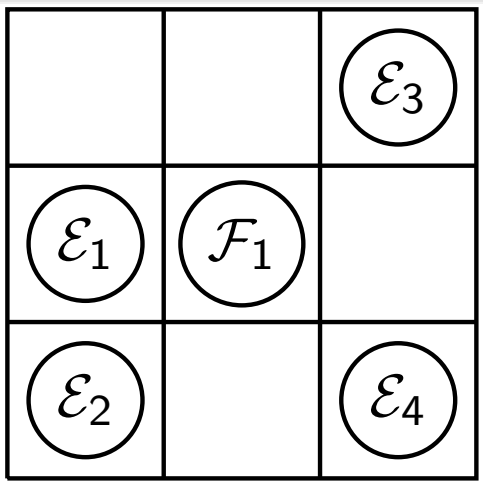
Behavior of an **Eater** $\mathcal{E}$:

- If there is at least one **Eater** nearby, $\mathcal{E}$ reproduces with it;
- else, if there is at least one **Food** nearby, $\mathcal{E}$ eats a part of it;
- else, it wanders.

Bias Example (3/3)

Behavior of an **Eater** $\mathcal{E}$:

- If there is at least one **Eater** nearby, $\mathcal{E}$ reproduces with it;
- else, if there is at least one **Food** nearby, $\mathcal{E}$ eats a part of it;
- else, it wanders.



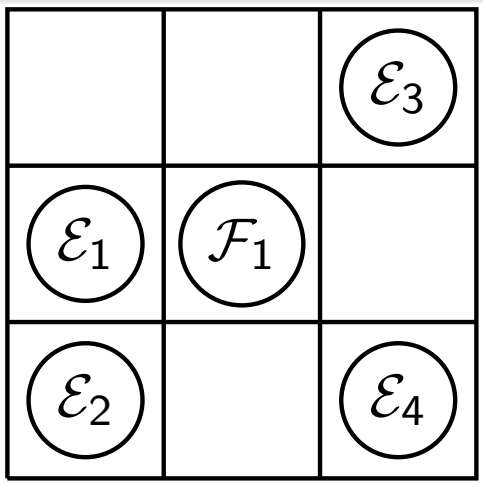
Particular setting of the environment

Agents order : $\mathbb{O}_t = \{\mathcal{E}_1, \mathcal{E}_2, \mathcal{F}_1, \mathcal{E}_3, \mathcal{E}_4\}$

Bias Example (3/3)

Behavior of an **Eater** $\mathcal{E}$:

- If there is at least one **Eater** nearby, $\mathcal{E}$ reproduces with it;
- else, if there is at least one **Food** nearby, $\mathcal{E}$ eats a part of it;
- else, it wanders.



Particular setting of the environment

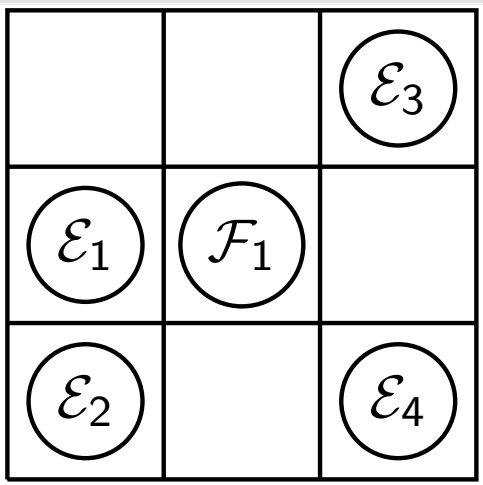
Agents order : $\mathbb{O}_t = \{\mathcal{E}_1, \mathcal{E}_2, \mathcal{F}_1, \mathcal{E}_3, \mathcal{E}_4\}$
$\mathcal{E}_1$ **perceived affordances** :

- *reproduce* with $\mathcal{E}_2$;
- or *eat* a part of $\mathcal{F}_1$;
- or *wander*.

Bias Example (3/3)

Behavior of an **Eater** $\mathcal{E}$:

- If there is at least one **Eater** nearby, $\mathcal{E}$ reproduces with it;
- else, if there is at least one **Food** nearby, $\mathcal{E}$ eats a part of it;
- else, it wanders.



Particular setting of the environment

Agents order : $\mathbb{O}_t = \{\mathcal{E}_1, \mathcal{E}_2, \mathcal{F}_1, \mathcal{E}_3, \mathcal{E}_4\}$
Performed Actions :

- $\mathcal{E}_1$ *reproduces* with $\mathcal{E}_2$

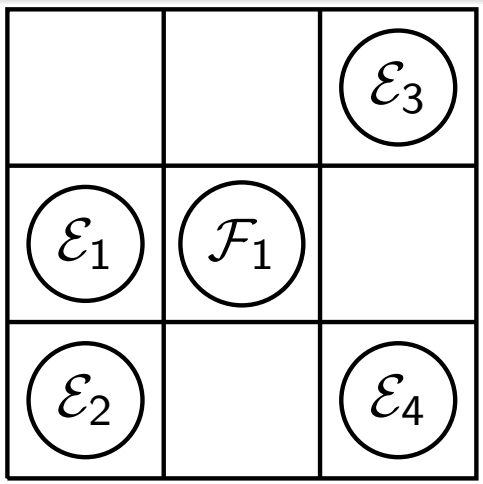
$\mathcal{E}_2$ **perceived affordances** :

- *reproduce* with $\mathcal{E}_1$;
- or *eat* a part of $\mathcal{F}_1$;
- or *wander*.

Bias Example (3/3)

Behavior of an **Eater** $\mathcal{E}$:

- If there is at least one **Eater** nearby, $\mathcal{E}$ reproduces with it;
- else, if there is at least one **Food** nearby, $\mathcal{E}$ eats a part of it;
- else, it wanders.



Particular setting of the environment

Agents order : $\mathbb{O}_t = \{\mathcal{E}_1, \mathcal{E}_2, \mathcal{F}_1, \mathcal{E}_3, \mathcal{E}_4\}$
Performed Actions :

- $\mathcal{E}_1$ *reproduces* with $\mathcal{E}_2$
- $\mathcal{E}_2$ *reproduces* with $\mathcal{E}_1$

$\mathcal{E}_1$ and $\mathcal{E}_2$ reproduce twice !

Bias Example (3/3)

Behavior of an **Eater** $\mathcal{E}$:

- If there is at least one **Eater** nearby, $\mathcal{E}$ reproduces with it;
- else, if there is at least one **Food** nearby, $\mathcal{E}$ eats a part of it;
- else, it wanders.

Underlying implicit choice

An agent that participates in an interaction may participate in any other interaction.

This error exists in [Epstein & Axtell 96] !!

Main Issue

Solution of this particular problem :

- If an **Eater** *reproduces*, it cannot participate to further reproduce interactions;
- If **Food** is *eaten* (the target of *eat*), it can still be *eaten* by other **Eaters**.

The solution seems obvious, but no design methodolgy specifies it clearly.

A generic solution requires :

- To handle agents according to the nature of the interaction;

Nowadays, no simulation frameworks or methodologies do specify precisely this point.

Outline

Preliminary Specification

As for any simulation, the modeler has to define how to represent time :

- Discrete Asynchronous
- Discrete Synchronous
- Continuous

For each time representation, an interaction executes during a time interval :

Discrete : the interval is implicitly the duration of a time step;

Continuous : the interval is explicitely defined by the modeler.

Preliminary Specification

For each time representation, an interaction executes during a time interval :

Discrete : the interval is implicitly the duration of a time step;

Continuous : the interval is explicitely defined by the modeler.

Simultaneous interactions

Two interactions are considered as simultaneous if the intersection of their time interval is not empty.

An Interaction Classification

To handle simultaneous interaction, we propose to give to any interaction a class, that represent different reccurent patterns used to handle agents according to interactions, among :

- Exclusive Interaction
- Parallel Interaction
- Systematic Interaction

This class provides an answer to the question :

> *"If an agent already participates in an interaction of $\mathcal{I}_1$ class, is it still able to participate in an interaction of $\mathcal{I}_2$ class ?"*

Summary of relationships

		Exclusive		Parallel		Systematic	
		S	T	S	T	S	T
Exclusive	S				X		X
	T				X		X
Parallel	S				X		X
	T	X	X	X	X	X	X
Systematic	S	X	X	X	X	X	X
	T	X	X	X	X	X	X

Figure: Summary of the interaction classes an agent can still participate in after participating in a particular interaction.

Link Between Interaction Class and Activation Unit

		Exclusive		Parallel		Systematic	
		S	T	S	T	S	T
Exclusive	S				X		X
	T				X		X

If an agent $\mathcal{A}$ is the **target** of an exclusive interaction then, until the interaction finishes :

- it cannot be the source of any other interaction : the Activation Unit will not select $\mathcal{A}$;

Link Between Interaction Class and Activation Unit

		Exclusive		Parallel		Systematic	
		S	T	S	T	S	T
Exclusive	S				X		X
	T				X		X

If an agent $\mathcal{A}$ is the **target** of an exclusive interaction then, until the interaction finishes :

- it cannot be the source of any other interaction : the Activation Unit will not select $\mathcal{A}$;
- it cannot be the target of any other exclusive interaction : other agents will not perceive in their affordances exclusive interactions with $\mathcal{A}$ as target.

Conclusion (1/2)

Conclusion :

- We want to build a generic and domain independent methodology and framework;
- It requires to identify :
  - the functional units underlying the architecture of any simulation;
  - for each unit a set of parameters that have to be precisely specified;
- We provided in this paper :
  - the identification proposal of those units;
  - the specification of one of the parameters of the Activation Unit
- Without the explicit specification of this parameter, a simuation is likely to be biased.

Conclusion (2/2)

This interaction-oriented methodology (IODA) and framework (JEDI) we built :

- make all these units explicit;
- reify them as software entities;
- provide a fine setting of implementation choices as parameters of the simulation core and interaction ontology;

see `http://www.lifl.fr/SMAC/projects/ioda`

Work in progress :

- Refine the functional decomposition;
- Continue the study of the Activation Unit;
- Study implementation choices in the Selection Unit[kubera, IAT'08];
- Complete the set of parameters proposed in IODA and JEDI.

Any questions ?