Sofware Architecture for Drone Simulation in 3D

EuSarlCon 2019 - May 2nd 2019

Dr. Alexandre LOMBARD Yazan MUALLA
Prof.Dr. Stéphane GALLAND Dr. Jocelyn BUISSON

Université de Technologie de Belfort Montbéliard - UTBM
90010 Belfort cedex, France - http://www.multiagent.fr

UBFC
UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ

OUTLINE

1 Context

2 SARL and AirSim

3 Introducing a framework for agent-based simulation

4 Defining Collision Avoidance Behavior

5 Results and Performance Benchmarks

OUTLINE

1 Context

2 SARL and AirSim

3 Introducing a framework for agent-based simulation

4 Defining Collision Avoidance Behavior

5 Results and Performance Benchmarks

DRONE IN SMART CITIES

1. Security and Privacy: How to guarantee the security and confidentiality of information captured by drones?
2. Safety: How to guarantee the continuous and safe operation of drones?
3. Traffic Regulation: What are the regulations to implement? At what level (international, EU, national, local)?
4. Traffic Management: How to control the air traffic and its fluidity?
5. Measuring the impact: of drone introduction on socio-economic and environmental aspects .

DRONE IN SMART CITIES

1. Security and Privacy: How to guarantee the security and confidentiality of information captured by drones?
2. Safety: How to guarantee the continuous and safe operation of drones?
3. Traffic Regulation: What are the regulations to implement? At what level (international, EU, national, local)?
4. Traffic Management: How to control the air traffic and its fluidity?
5. Measuring the impact: of drone introduction on socio-economic and environmental aspects.

OUTLINE

1 Context

2 SARL and AirSim

3 Introducing a framework for agent-based simulation

4 Defining Collision Avoidance Behavior

5 Results and Performance Benchmarks

Multiagent System in SARL

A collection of agents interacting together in a collection of shared distributed spaces.

4 main concepts

- Agent
- Capacity
- Skill
- Space

3 main dimensions

- Individual:: the Agent abstraction (Agent, Capacity, Skill)
- Collective:: the Interaction abstraction (Space, Event, etc.)
- Hierarchical:: the Holon abstraction (Context)

SARL: a general-purpose agent-oriented programming language. Rodriguez, S., Gaud, N., Galland, S. (2014) Presented at the The 2014 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IEEE Computer Society Press, Warsaw, Poland. (Rodriguez et al., 2014)

`http://www.sarl.io`

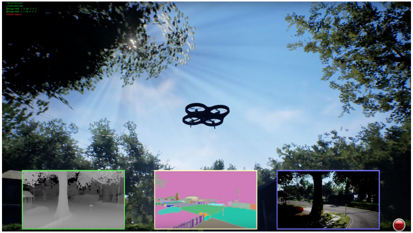
AIRSIM FRAMEWORK

AirSim is:

an open-source simulation platform developed by Microsoft

Major Characteristics

- Focuses on drone and car simulation.
- Manages the physics of the world.
- Simulates sensors, e.g. laser range finder.
- Provides an API to allow sending control commands to the vehicles.



https://github.com/Microsoft/AirSim

OUTLINE

1 Context

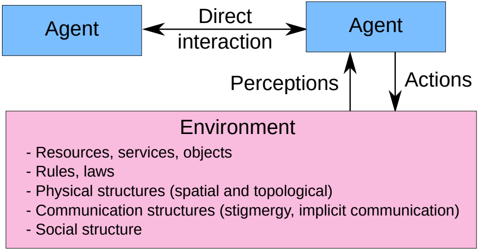
2 SARL and AirSim

3 Introducing a framework for agent-based simulation

4 Defining Collision Avoidance Behavior

5 Results and Performance Benchmarks

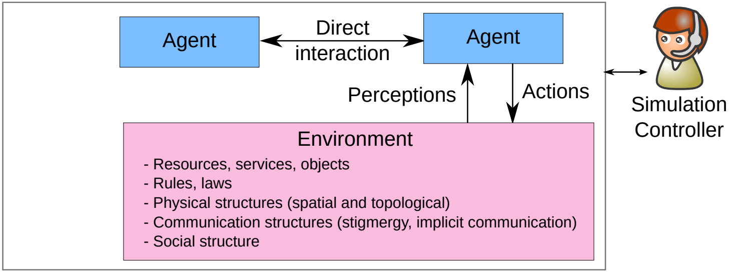
GENERAL AGENT-BASED SIMULATION ARCHITECTURE



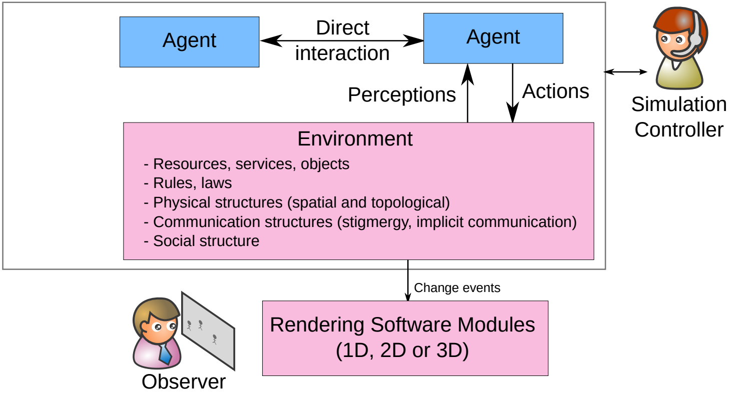
Environment

- Resources, services, objects
- Rules, laws
- Physical structures (spatial and topological)
- Communication structures (stigmergy, implicit communication)
- Social structure

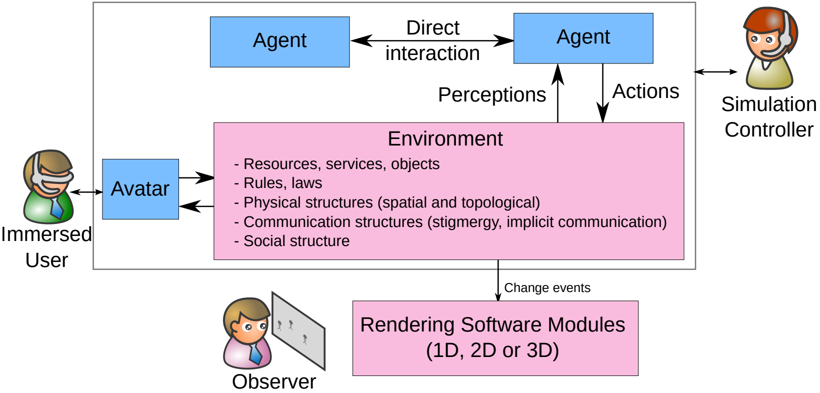
GENERAL AGENT-BASED SIMULATION ARCHITECTURE


Agent
Direct interaction
Agent
Perceptions
Actions
Simulation Controller
Environment
- Resources, services, objects
- Rules, laws
- Physical structures (spatial and topological)
- Communication structures (stigmergy, implicit communication)
- Social structure

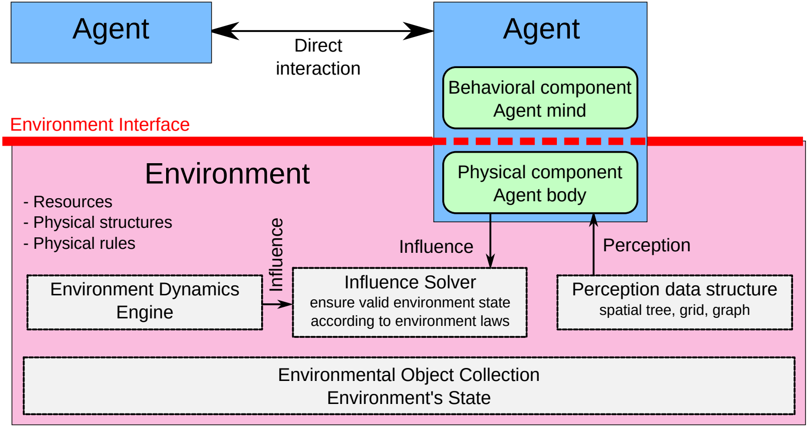
GENERAL AGENT-BASED SIMULATION ARCHITECTURE


Agent
Direct interaction
Agent
Perceptions
Actions
Simulation Controller
Environment
- Resources, services, objects
- Rules, laws
- Physical structures (spatial and topological)
- Communication structures (stigmergy, implicit communication)
- Social structure
Change events
Rendering Software Modules (1D, 2D or 3D)
Observer

GENERAL AGENT-BASED SIMULATION ARCHITECTURE


Agent
Direct interaction
Agent
Perceptions
Actions
Simulation Controller
Environment
- Resources, services, objects
- Rules, laws
- Physical structures (spatial and topological)
- Communication structures (stigmergy, implicit communication)
- Social structure
Avatar
Immersed User
Change events
Rendering Software Modules (1D, 2D or 3D)
Observer

SITUATED ENVIRONMENT MODEL


Agent

Direct interaction

Agent

Behavioral component
Agent mind

Environment Interface

Physical component
Agent body

Environment

- Resources
- Physical structures
- Physical rules

Influence

Influence

Perception

Environment Dynamics Engine

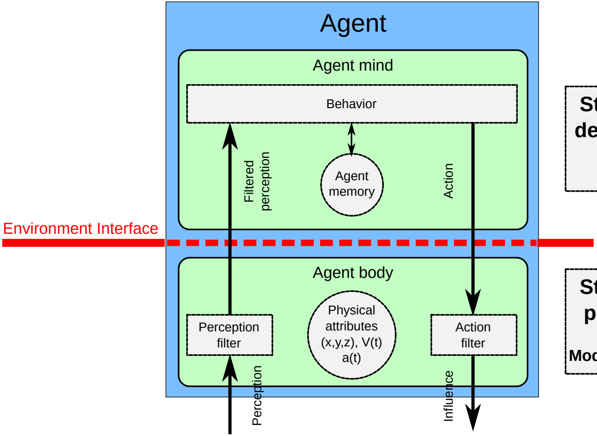
Influence Solver
ensure valid environment state according to environment laws

Perception data structure
spatial tree, grid, graph

Environmental Object Collection
Environment's State

Galland et al. (2009)

BODY-MIND DISTINCTION



State variables of the decisional component
Readable/modifiable only by the agent

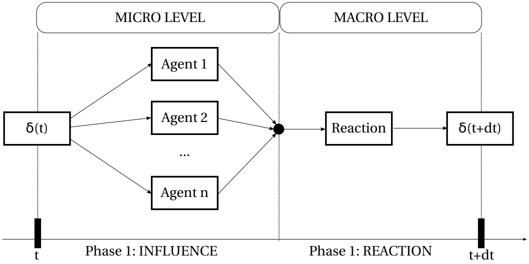
State variables of the physical component
Readable by the agent
Modifiable by the environment

Galland et al. (2009)

SIMULTANEOUS ACTIONS: INFLUENCE-REACTION

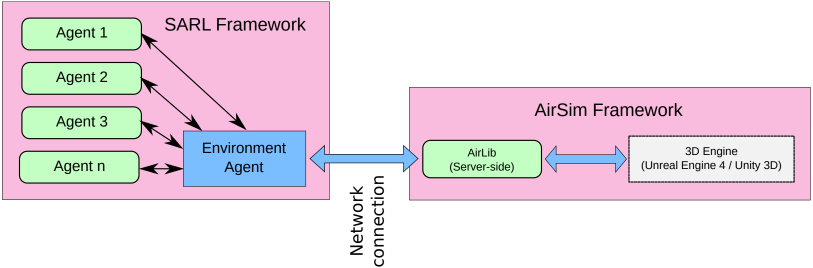
How to support simultaneous actions from agents?

1. An agent does not change the state of the environment directly.
2. Agent gives a state-change expectation to the environment: the influence.
3. Environment gathers influences, and solves conflicts among them for obtaining its reaction.
4. Environment applies reaction for changing its state.



Michel (2007)

SARL-AIRSIM ARCHITECTURE



ROLE OF THE ENVIRONMENT AGENT

1. Link agents to vehicles:
   - Receive the perceptions of each vehicle and dispatch them to linked agents
   - Collect and transmit the influences from the agents to AirSim

2. Manage the simulation schedule

BODY MANAGEMENT CAPACITY

- The agent has the capacity to use its body.
- The body supports the interactions with the environment.

```
event Perception {
  val object : Object
  val relativePosition : Vector
}

capacity EnvironmentInteraction {
  def moveTheBody(motion : Vector)
  def move(object : Object,
    motion : Vector)
  def executeActionOn(object :
     UUID,
    actionName : String,
    parameters : Object*)
}

skill PhysicBody implements
    EnvironmentInteraction {

  val env : UUID
  val body : UUID
  def moveTheBody(motion:Vector) {
    move(this.body, motion)
  }
  def move(object : UUID,
      motion : Vector) {
    emit(new MotionInfluence(
      object, motion))
      [UUID == this.env]
  }

  def executeActionOn(
      object : UUID,
      actionName : String,
      parameters : Object*) {
    emit(new ActionInfluence(
      object, actionName,
      parameters))
      [UUID == this.env]
  }
}
```

Galland et al. (2015)

OUTLINE

1 Context

2 SARL and AirSim

3 Introducing a framework for agent-based simulation

4 Defining Collision Avoidance Behavior

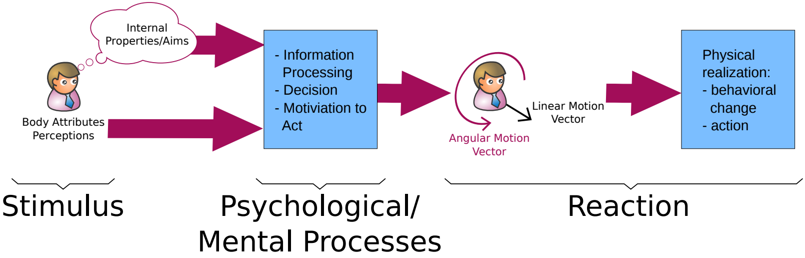
5 Results and Performance Benchmarks

MODELING BEHAVIORAL CHANGES

- K. Lewin. “Social fields (Social forces).” Field Theory in Social Science, Harper & Brothers, New York, 1951.

- “A sensory stimulus causes a behavioral reaction that depends on the personal aims; and
- It is chosen from a set of behavioral alternatives with the objective of utility maximization.”

SCHEMATIC REPRESENTATION



Internal Properties/Aims

Body Attributes Perceptions

- Information Processing
- Decision
- Motiviation to Act

Angular Motion Vector

Linear Motion Vector

Physical realization:
- behavioral change
- action

Stimulus

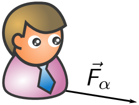
Psychological/ Mental Processes

Reaction

Lewin (1951)

SOCIAL FORCE FOR DRONE

- The systematic temporal changes $d\vec{w}_\alpha/dt$ of the preferred velocity $\vec{w}_\alpha(t)$ of a drone α are described by a vectorial quantity $\vec{F}_\alpha(t)$.



- This force represents the effect of the environment (e.g. other drones or borders) on the behavior of the described drone.
- It is a quantity that describes the concrete motivation to act.
- One can say that a drone acts as if it would be subject to external forces.

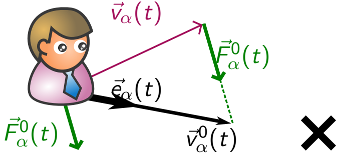
Helbing and Molnar (1997)

ACCELERATION TERM

- If a drone's motion is not disturbed, it will walk into the desired direction $\vec{r}_\alpha(t)$ with a certain desired speed v_α^0.

- A deviation of the actual velocity $\vec{v}_\alpha(t)$ to the desired velocity $\vec{v}_\alpha^0(t) = v_\alpha^0 \vec{e}_\alpha(t)$ due to necessary deceleration processes or avoidance processes leads to a tendency to approach $\vec{v}_\alpha^0(t)$ again within a certain relaxation time τ_α:

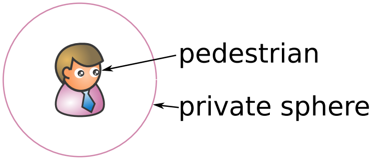
$$\vec{F}_\alpha^0(\vec{v}_\alpha, v_\alpha^0 \vec{e}_\alpha) = \frac{1}{\tau_\alpha}(v_\alpha^0 \vec{e}_\alpha - \vec{v}_\alpha)$$



Helbing and Molnar (1997)

PRIVATE SPHERE AND REPULSIVE EFFECTS

- The motion of a drone α is influenced by other drones.
- It keeps a certain distance from other drones that depends on the drone density and the desired speed v_α^0.
- The private sphere of each drone, which can be interpreted as territorial effect (Scheflen and Ashcraft, 1976), plays an essential role.



pedestrian

private sphere

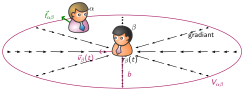
- A drone normally feels increasingly incomfortable the closer another drone β is inside its private sphere.
- This results in repulsive effects of β that can be represented by vectorial quantities $\vec{f}_{\alpha\beta}$.

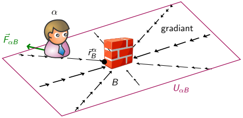
Helbing and Molnar (1997)

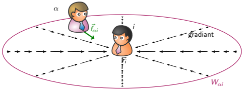
SOCIAL FORCES

The total motivation of a drone is affected by

- Its acceleration term $\vec{F}^0_\alpha(\vec{v}_\alpha, v^0_\alpha \vec{e}_\alpha)$
- The repulsion from mobile objects
- The repulsion from immobile objects
- The attraction to other objects







Helbing and Molnar (1997)

SOCIAL FORCE-MODEL

- The total motivation for the drone α is:

$$\begin{aligned}\vec{F}_\alpha(t) = \;& \vec{F}^0_\alpha(\vec{v}_\alpha, v^0_\alpha \vec{e}_\alpha) && \text{acceleration term}\\ & + \textstyle\sum_\beta \vec{F}_{\alpha\beta}(\vec{e}_\alpha, \vec{r}_\alpha - \vec{r}_\beta) && \text{mobile object repulsion term}\\ & + \textstyle\sum_B \vec{F}_{\alpha B}(\vec{e}_\alpha, \vec{r}_\alpha - \vec{r}^\alpha_B) && \text{immobile object repulsion term}\\ & + \textstyle\sum_i \vec{F}_{\alpha i}(\vec{e}_\alpha, \vec{r}_\alpha - \vec{r}_i, t) && \text{attraction term}\end{aligned}$$

- The social force model becomes:

$$\frac{d\vec{w}_\alpha}{dt} = \vec{F}_\alpha(t) + \mathit{fluctuations}$$

- The *fluctuation* term takes into account random variations of the behavior.

Helbing and Molnar (1997)

SLIDING FORCES

Problems with the Social Force Model

1. Difficult to define the attraction term: what is the attraction length, regarding the repulsive terms?
2. It may be difficult to reproduce the smooth trajectories that are followed by real Humans.

Introduction of sliding forces that hide the attractive force, and make the trajectory smoother.

Buisson et al. (2013)

SLIDING FORCE MODEL

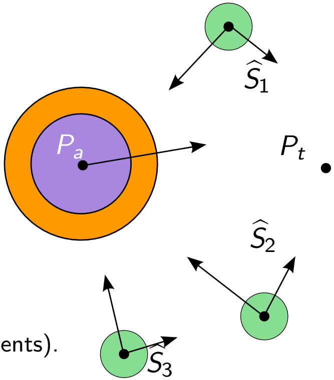
The force to apply to each agent is:

$$\vec{F_a} = \vec{F} + w_a.\delta_{\|\vec{F}\|}.(\widehat{p_t - p_a})$$

$$\vec{F} = \sum_{i \in M} U(t_c^i) \cdot \widehat{S}_i$$

where:

- $\vec{F}$ is the collision-avoidance force.
- $\hat{S}_i$ is a sliding force.
- t_c^i is the time to collision to object i.
- $U(t)$ is a scaling function of the time to collision.
- M is the set objects around (including the other agents).
- w_a is the weight of the attractive force.
- $\delta_x g$ is g if $x \leq 0$, 0 otherwise.



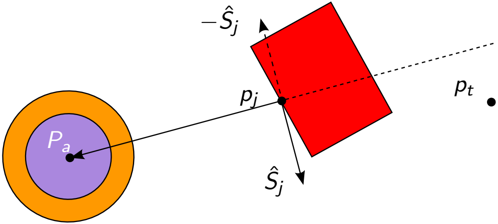
Buisson et al. (2013)

SLIDING FORCE

- The sliding force $\vec{S_i}$ is:

$$\vec{s_j} = (p_j - p_a) \times \hat{y}$$

$$\hat{S_j} = \text{sign}(\vec{s_j} \cdot (\vec{p_t} - p_a)).\widehat{s_j}$$



- $\hat{y}$ is the vertical unit vector.

Buisson et al. (2013)

SCALING THE SLIDING FORCE

- How to scale $\hat{S}_j$ to obtain the repulsive force?
- Many force-based models use a monotonic decreasing function of the distance to an obstacle.
- But it does not support the velocity of the agent.
- Solution: Use time-based force scaling function.

$$U(t) = \begin{cases} \frac{\sigma}{t^{\phi}} - \frac{\sigma}{{t_{max}}^{\phi}} & \text{if } 0 \leq t \leq t_{max} \\ 0 & \text{if } t > t_{max} \end{cases}$$

where:
- t is the estimated time to collision.
- t_{max} is the the maximum anticipation time.
- σ and ϕ are constants, such that $U(t_{max}) = 0$

Buisson et al. (2013)

OUTLINE

1 Context

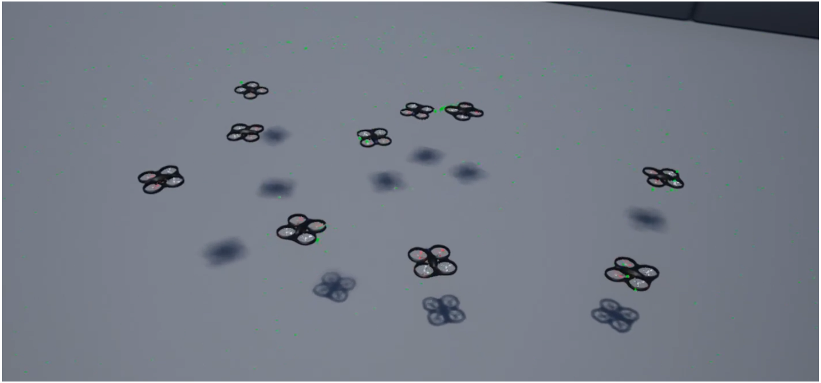
2 SARL and AirSim

3 Introducing a framework for agent-based simulation

4 Defining Collision Avoidance Behavior

5 Results and Performance Benchmarks

FORCE-BASED SIMULATION OF DRONES IN AIRSIM



PERFORMANCE BENCHMARKS

- Simulations can run in real-time with up to 50 drones (30 FPS on a good PC)
- The current bottleneck is AirSim
- It is still possible to increase the amount of drones (but it will not be in real-time anymore)
- There is on-going work made to improve AirSim efficiency for multi-agents simulations

Thank you for your attention. . .

Appendix

GITHUB REPOSITORIES

- http://github.com/alexandrelombard/sarl-airsim-interface
- http://github.com/alexandrelombard/airsim-jvm-api

OUTLINE

Bibliography (#1)

Buisson, J., Galland, S., Gaud, N., Yasar, Gonçalves, M., and Koukam, A. (2013). Real-time collision avoidance for pedestrian and bicyclist simulation: a smooth and predictive approach. In *the 2nd International Workshop on Agent-based Mobility, Traffic and Transportation Models, Methodologies and Applications (ABMTRANS)*, Halifax, Nova Scotia, Canada. Procedia Computer Science, Elsevier.

Galland, S., Balbo, F., Gaud, N., Rodriguez, S., Picard, G., and Boissier, O. (2015). Contextualize agent interactions by combining social and physical dimensions in the environment. In Demazeau, Y., Decker, K., De la prieta, F., and Bajo perez, J., editors, *Advances in Practical Applications of Agents, Multi-Agent Systems, and Sustainability: The PAAMS Collection. Lecture Notes in Computer Science 9086.*, pages 107–119. Springer International Publishing.

Galland, S., Gaud, N., Demange, J., and Koukam, A. (2009). Environment model for multiagent-based simulation of 3D urban systems. In *the 7th European Workshop on Multiagent Systems (EUMAS09)*, Ayia Napa, Cyprus. Paper 36.

Helbing, D. and Molnar, P. (1997). Self-organization phenomena in pedestrian crowds. *Self-organization of complex structures: from individual to collective dynamics*, pages 569–577.

Lewin, K. (1951). Social fields (social forces). *Field Theory in Social Science*.

Michel, F. (2007). The IRM4S model: the influence/reaction principle for multiagent based simulation. In *AAMAS '07: Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, pages 1–3, New York, NY, USA. ACM.

Rodriguez, S., Gaud, N., and Galland, S. (2014). SARL: a general-purpose agent-oriented programming language. Warsaw, Poland. IEEE Computer Society Press.

Scheflen, A. and Ashcraft, N. (1976). *Human Territories: How We Behave in Space-Time*. Prentice-Hall, Englewood Cliffs.