Multi-level model of a collective motion phenomenon

Benjamin Camus, Christine Bourjot, Vincent Chevrier
Université de Lorraine, CNRS, LORIA UMR 7503, Vandoeuvre-lès-Nancy, F-54506, France.
INRIA, Villers-lès-Nancy, F-54600, France.
benjamin.camus@loria.fr christine.bourjot@loria.fr vincent.chevrier@loria.fr

Overview

Purpose

This multi-model (i.e. a model composed of interacting submodels) corresponds to a multi-level representation of a collective motion phenomenon. In this example, we consider individuals evolving in a toric space and forming groups. A group is defined here as a set of close enough individuals with similar orientation. This model was designed to study the impact of the mutual influences between individuals and groups on the formation of groups in a collective motion. The scientific description of this model can be found in (Camus et al., TBP).

The multi-model is composed of two multi-agent submodels m and M respectively representing the phenomenon from a micro (individuals) and a macro (groups) perspectives. Each of these submodels is already implemented in separated NetLogo simulator instances. Using the AA4MM (Agent & Artifact for Multi-Modeling) metamodel (Siebert et al. 2010), the multi-model is described as an Agent & Artifact (A&A) system (Ricci et al., 2007).

We are then considering agents at two levels:

- The agents at the models level represent the entities of the system we want to model. These entities correspond to individuals and groups.
- The agents at the multi-model level manage the multi-model execution. Such agents are named m-agents.

At the multi-model level, we describe the interaction processes between the models, and, therefore, the influences between individuals and groups, thanks to the concept of artifact. We can then compare the simulation results (i.e. number of formed groups as a function of time) of different interaction patterns between models.

State Variables and Scales

As mentioned in the 'Purpose' section, the multi-model is represented as an A&A system. Following this paradigm, the agents' environment is composed of artifacts. Each of these artifacts is a passive computational entity providing services to the agent.

AA4MM, relies on three concepts to describe a multi-level multi-model:

- Each submodel is associated with an autonomous m-agent. This m-agent manages the model and is in charge of the interactions of this model with the other ones. Two m-agents

compose the multi-model:
    - The micro m-agent manages the model m
    - The macro m-agent manages the model M
- A coupling artifact reifies each interaction between the two m-agents. In order to manage multi-level interaction, two kinds of coupling artifacts compose the multi-model:
    - Upward artifact ensures the translation from model m to model M: individuals are translated into groups.
    - Downward artifact ensures the translation from model M to model m: groups are translated into individuals.
- Finally, the interface artifact reifies the interactions between an m-agent and its model.

The states of these entities are described by the following low-level state variables.

M-agent

Name	**Description**
Current_time	The current simulation time of the m-agent's model.
Max-Sim-Time	The maximum simulation time of the multi-model's simulation

Upward Coupling Artifact

Name	**Description**
Event queue	The queue of the simulation data dropped by an m-agent
Proximity_threshold	Used to detect groups based on individuals' positions: defines what « a set of close enough individuals » means for a group.
Orientation_threshold	Used to detect groups based on individuals' orientation: define what «a set of individuals with similar orientation » means for a group.

Downward Coupling Artifact

Name	**Description**
Event queue	The queue of the simulation data dropped by an m-agent

Process Overview and Scheduling

In the multi-model, the environment (i.e. the artifacts and the submodels) and each m-agent correspond to distinct processes. Each submodel is considered as an event-based model.

The behavior of each agent corresponds to a simplified version of the Chandy Misra Bryant (CMB) algorithm (Figure 1). This algorithm enables the synchronization of the m-agents' submodels in the simulated time.

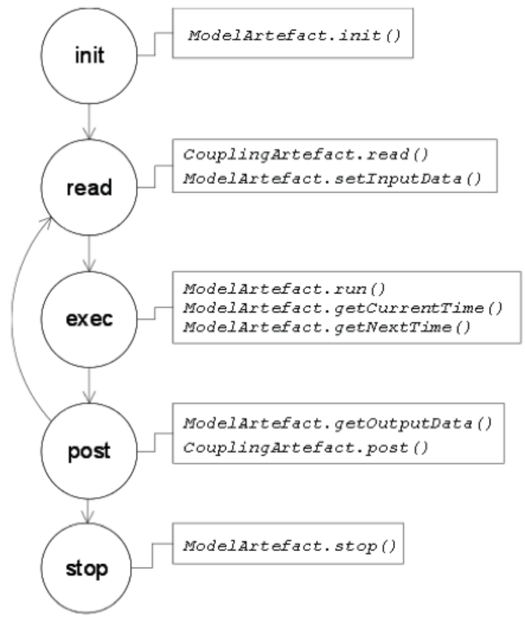

Figure 1. The m-agent behavior (taken from (Siebert et al. 2010))

The m-agents control their submodels by using their model artifact. The m-agents exchange their models' simulation data through the coupling artifacts. Each coupling artifacts works like a mailbox: a m-agent drops its submodel's output simulation data, and another m-agent picks up its submodel's input simulation data. Each coupling artifact uses operations to transform output simulation data into input simulation data.

The upward artifact detects groups based on individuals' states using a three dimensions (with dimensions corresponding to x,y individuals positions, and orientation) version of the optics algorithm (Ankerst et al. 1999). This function takes two parameters: proximity and orientation thresholds. The upward artifact sets detected groups' attributes by aggregating micro data as follow: position is the gravity center of the cluster, orientation corresponds to the average direction of its constituent individuals and size refers to the dispersion. The downward artefact translate groups' movements into individual movements: each individual in a group moves with the same vector that its group.

Design Concepts

Emergence

The multi-model simulation is performed thanks to the co-evolution of the m-agent's submodels and the exchanges of simulation data between them.

Adaptation

The m-agents wait for valid input simulation data from their coupling artifacts before processing a time step of their submodels.

Fitness/Objectives

The goal of each m-agent is to perform the simulation of its submodels until the maximum simulation time.

Interaction

The m-agents interactions consist of exchanging their submodels' simulation data. These data are exchanged in an indirect way through the coupling artifacts.

Stochasticity

The multi-models evolution is a purely determinist process. However, the state of submodel *m* is initialized randomly.

Observation

The evolution of each submodels' states is graphically plotted in their simulation software. Each m-agents can collect data from its submodel's states and write it into output files.

Details

Initialisation

Before the beginning of the multi-model's simulation, each m-agent follows the steps:

1) It initializes its submodel's simulation software.
2) It shares its submodel initial states with other m-agents.
3) It parameters its submodel based on the initial states of the other m-agents' submodels.

Submodels

The micro submodel m corresponds to the collective motion model of Netlogo (Wilensky 1998). It represents moving individuals. Each individual is defined by an identifier, a position, an orientation and a color. The behavior of an individual consists of three rules:

- Trying to avoid collisions with obstacles (others individuals),
- Moving closer to others individuals,
- Following other individuals.

This submodel assumes that an individual doesn't have a destination. Therefore, in the absence of others individuals, it goes straightforward. The individuals move at constant speed and have initially the same color that remains unchanged during movements.

The macro submodel M is implemented in the Netlogo simulator. M represents moving groups. A group is defined as a set of "close enough" individuals with similar orientations. In the macro model, a group has an identifier, a position, an orientation, a size and a color (different for each group). We assume here that the behavior of a group is the same that the behavior of an individual in m: a group tries to move closer and to follow other groups while trying to avoid collisions with obstacles.

References

ANKERST M., Breunig M., Kriegel H-P., Sander J. (1999). "OPTICS: Ordering Points to Identify the Clustering Structure". ACM SIGMOD International Conference on Management of Data. ACM Press. pp. 49–60.

CAMUS B., Bourjot C., Chevrier V. (TBP). “Considering a Multi-Level Model as a Society of Interacting Models: Application to a Collective Motion Example". Journal of Artificial Societies and Social Simulation (JASSS).

RICCI A., Viroli M., Omicini A. (2007). “Give Agents their Artifacts: the A&A Approach for Engineering Working Environments in MAS.” In AAMAS '07: Proc. of the 6th international joint conference on Autonomous agents and multiagent systems, pp601-603, New York, USA.

SIEBERT J., Ciarletta L., Chevrier V. (2010). “Agents and Artifacts for Multiple Models Co-evolution. Building Complex System Simulation as a set of Interacting Models.” In Proceedings of the International Conference on Autonomous Agents and Multiagent Systems, in proc. of AAMAS 2010, May 10, 2010, Toronto, Canada, pp509-516.

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.