

SIMEO

Octavian Voicu, Andrei Ciortea – Msc. AI

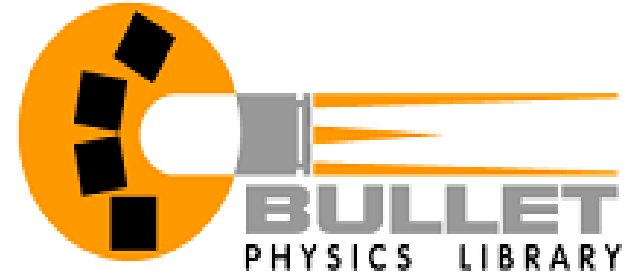
Bogdan Ghit – Msc. PDCS

Context

- **SIM**ulation Architecture for the Study of
Emergence and Auto-
Organization
- Generic framework designed to facilitate the **study** of **interactions** between large numbers of agents in a **simulated virtual world**
- Exploits parallelism
- Several communication models
- Design with the Agent paradigm

Technologies

- Bullet Physics Library
- Message Passing Interface (MPI)
- Compiled Native Interface (CNI)
- Languages: C++, Java
- Tools: KDevelop, Cmake



System Architecture

- **MAIN CONTROLLER** - manages the other controllers + load balancing
 - creates controller for each environment zone
 - allows agents to join the platform
 - ensures communication with the environment
- **CONTROLLERS** - facilitate efficient communication between other entities living in the same zone
 - establishes virtual communication channels
 - relays percepts from environment to sensors
 - relays commands from actuators to environments
- **AGENTS**
 - SENSORS - monitor the environment
 - ACTUATORS - apply changes to the environment
 - CHANNELS - allow agents to communicate

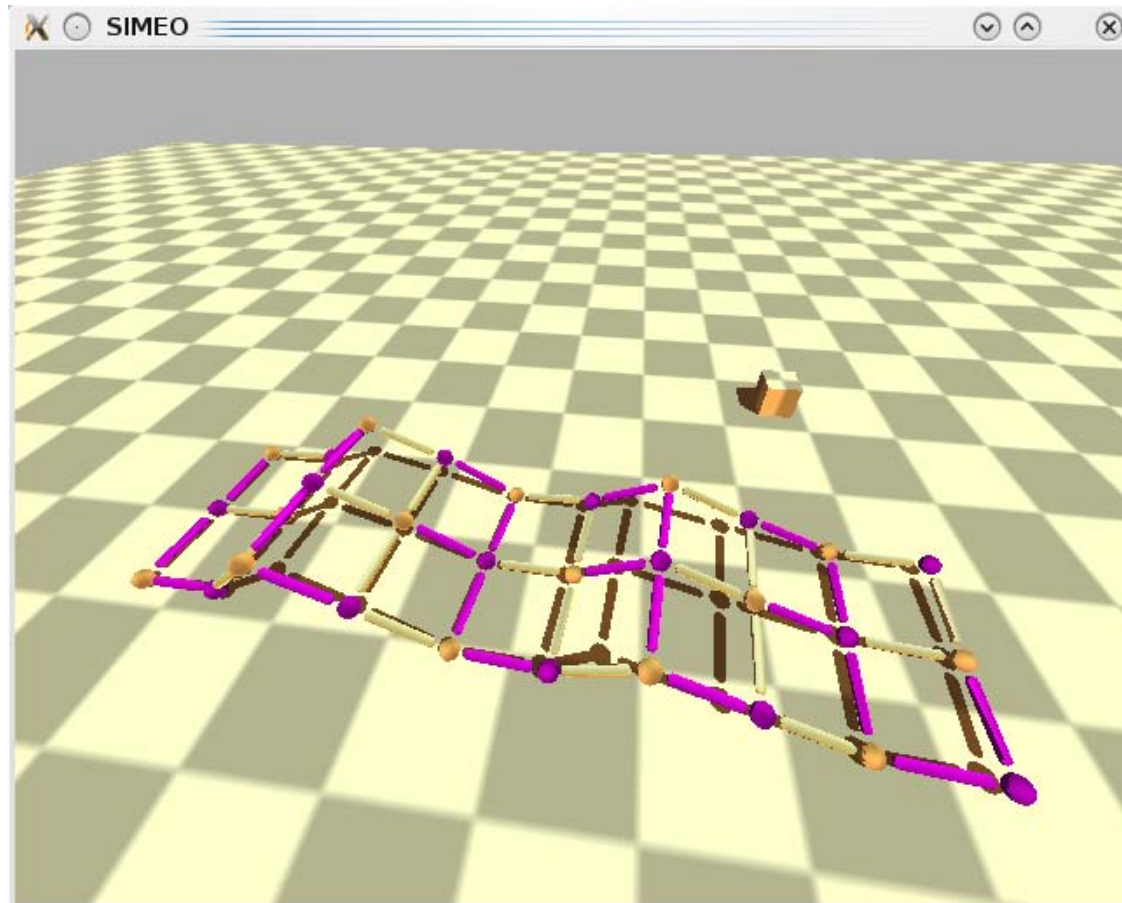
Communication Mechanisms

- MAIN CONTROLLER \leftrightarrow ENVIRONMENT
 - Currently runs on the same machine
 - Framework designed to allow running the Environment remotely (i.e. on a Cell Cluster)
- AGENTS \leftrightarrow ENVIRONMENT
 - Channels
 - Actuators
- AGENTS \leftrightarrow AGENTS
 - Communication Channels

Current Implementation

- Framework: implemented in C++, extensible
- Environment: simulated using the Bullet Physics Library
- Channel implementation: OpenMPI
- Remote GUI protocol: TCP based
- Agents: support for Sensors, Actuators, Communication Channels
- Sensors/Actuators: 2DOF-constraint based

Simeo GUI and the Lazy Worm



Contributions

Octavian

- Original Idea & Vision
- Project Setup
- Technology Testing and POCs
- *The Lazy Worm Walk*
- Various Bits of Code and Tweaks

Bogdan

- Framework Implementation
- MPI Communication Code
- Multi-threaded Local GUI
- Communication Channels
- Synchronization

Team Effort

- Brainstorming
- Design of the Architecture

Andrei

- Agent Code
- Work on Sensors and Actuators
- Remote GUI for Real-Time Execution on a Cluster
- TCP Communication Code
- Documentation

Quest ions?