**Noname manuscript No.**
(will be inserted by the editor)

# Environment orientation

## An approach to the simulation of complex systems

**Tim Hoverd · Susan Stepney**

Received: date / Accepted: date

**Abstract** A naïve implementation of a complex system simulation with its plethora of interacting agents would be to represent those interactions as direct communications between the agents themselves. Considerations of the real world that a complex system inhabits shows that agent interactions are actually mediated by the environment within which they are embedded and which embodies facilities used by the agents. This suggests an "environment oriented" simulation architecture.

**Keywords** Complex systems · agent-based simulation · software architecture

Complex systems comprise of a number of agents that interact in some particular environment. The behaviour of any individual agent is relatively simple and local. A complex global behaviour emerges as a consequence of the interaction between a large number of such agents in a particular environment.

A naïve approach to the computational simulation of such a system would represent agents as components of the simulation and would directly implement interactions between those components as simulations of the complex system interactions. Such an approach introduces many problems as a consequence of the inherently sequential nature of the underlying computing resources. In particular, deadlocks are common and must be resolved by introducing synchronisation mechanisms that are noticeably absent in the complex system itself.

Consideration of the real world shows that complex systems agents do not directly interact with each other, particularly in the simplistic way described. Rather, the agents interact with their environment and with the fields that are mediated by that environment. For example, agents *see* each other as a consequence of the passage of light through the environment, they hear each other by the pressurisation of the atmosphere, again mediated by the environment. That is, each agent only interacts with its environment, not with other agents.

The environment must then be seen as the mechanism that supports the fields in which the agents exist. The environment must also *embody* the physics of the real world

Tim Hoverd, Susan Stepney
Department of Computer Science, University of York, UK, YO10 5DD
E-mail: tim.hoverd@cs.york.ac.uk, susan@cs.york.ac.uk

in that sound pressure waves must attenuate over distance and deposited pheromones must evaporate.

An agent can be seen as placing into the environment state information that describes how that agent is seen externally. However, that *external state* may just be a view, or some other representation, of its *internal state*. An agent might know where it is in its environment, in which direction it is travelling and perhaps why it is travelling in that direction. Its external state, though, might just describe where it is and its direction of movement.

When another agent "looks" in the environment and sees the first agent's external state, the fidelity with which it sees that state is also a function of the environment. An agent might not be able to determine the range to another distant agent, just its direction and its proper motion. That is, rather than seeing the distant agent's position in a 3-dimensional space, it is limited to seeing the position in a 2-dimensional space; in this case on the surface of a sphere centred on the observing agent. The information provided by the environment is therefore embedded in a particular topology that reflects the fidelity of the observed agent's raw external state and the relationship, spatial or otherwise, between the two agents; observer and observee. It them becomes part of the role of the observing agent to combine information, perhaps ostensibly of the same form, but embedded in a number of different topologies, into its own model of the world, from which it must determine its future behaviour.

The concept of agents interacting merely with an environment immediately suggests an appropriate implementation architecture. Such an architecture would be based around some sort of repository with agents accessing that repository in an essentually transactional manner. Each, and every, agent would sequentially access information from the environment, use that information to update its internal state and then publicise its external state back to the environment. That would be a "query-oriented" approach. An alternative is a "subscription" approach where an agent would ask the environment to pass it information as it appeared, and was relevant to that agent. The latter approach is perhaps more representative of what happens in a complex system.

Experimentation with environmentally oriented implementations is in progress. The environment takes one of two forms in current implementations; as either a Linda tuple space or a relational database. Both implementations provide a suitable repository for state information and a mechanism for extracting information from the environment.

Future work will build more realistic simulations on the current platforms.

**Acknowledgements** Acknowledgements This work is part of the CoSMoS project, funded by EPSRC grant EP/E053505/1 and a Microsoft Research Europe PhD studentship.