Agent-based modelling of interdependent critical infrastructures

E. Casalicchio*, E. Galli and S. Tucci

Faculty of Engineering,
Department of Computer Science, Systems and Production,
University of Roma 'Tor Vergata',
Via del Politecnico 1, 00133 Roma, Italy
E-mail: emiliano.casalicchio@uniroma2.it
E-mail: galli@disp.uniroma2.it
E-mail: tucci@uniroma2.it
*Corresponding author

Abstract: Critical interdependent infrastructures are complex systems, that if damaged or disrupted can seriously compromise the welfare of our society. This research, part of the CRESCO project, deal with the problem of interdependent critical infrastructures analysis, proposing an agent-based modelling and simulation solution. The approach we put forward, named Federated-ABMS, relies on discrete agent-based modelling and simulation and federated simulation. Federated-ABMS provides a formalism to model compound complex systems, composed of interacting systems, as federation of interacting agents and sector specific simulation models. This paper describes the formal model as well it outlines the steps that characterise the Federated-ABMS methodology, here applied to a target system, composed of a communication network and of a power grid. Moreover we conclude the paper with a thorough discussion of implementation issues.

Keywords: agent-based modelling and simulation; critical infrastructure protection; interdependencies analysis; federated simulation.

Biographical notes: Emiliano Casalicchio is a Researcher at the University of Roma Tor Vergata. His research is mainly focussed on complex systems modelling and simulation (interdependencies analysis in critical infrastructures) and on distributed systems design, modelling and performance evaluation (web services, GRID computing and cloud computing). He is the author of about 60 publications in international journals and conferences, and he serves as a Reviewer for international journals and conferences (IEEE, ACM, Elsevier, . . .). He is and has been involved in many Italian Research project and EU Research projects (PERF, CRESCO, MIA, MOTIA, D-ASAP, MOBI-DEV, AEOLUS, DELIS, SMS, . . .).

Emanuele Galli is a PhD student at the University of Roma Tor Vergata. He received his Masters in Computer Science from the same university. He has worked for almost one year at Los Alamos National Laboratory. His

research interest includes distributed simulation, agent-based simulation and critical infrastructure modelling and analysis.

Salvatore Tucci is a Full Professor of Computer Engineering at the University of Roma Tor Vergata, Italy. He was a Researcher in Computer Science at the University of Pisa, a Visiting Scientist at IBM T.J. Watson Research Center, a Visiting Researcher at INRIA, University of Paris-Nord. His current research activities deal with performance evaluation of internet architecture and critical infrastructures modelling and simulation. In the last 30 years, he has done researches in the areas of modelling and performance evaluation, reliability, simulation of parallel and distributed systems. He has been responsible for several research projects and He is author of several scientific articles.

1 Introduction

Many researches on critical infrastructure protection are committed to solve the challenging problem of interdependencies modelling and analysis or more in general of modelling and simulation of critical interdependent infrastructures. While some research results are based on mathematical models (Rinaldi et al., 2001; Haimes and Jiang, 2001; Asavathiratham et al., 2001; Gursesli and Desrochers, 2003; Svendsen and Wolthusen, 2007; Zhang et al., 2005), other solutions rely on discrete simulation. An extended survey on simulation approach can be found in Paderson et al. (2006). In the specific, agent-based solution are proposed in Dudenhoeffer et al. (2006), Panzieri et al. (2004), Balducelli et al. (2005), Gianni et al. (2008), Casalicchio et al. (2007) and Galli et al. (2009), SimCIP (Beyer et al., 2007), DIESIS (EU Project DIESIS).

All these works are mainly focalised on the results obtained or achievable through agent-based simulation or on the architecture of the simulation framework. Nobody focalises the attention on how dependencies and interdependencies are modelled (or could be modelled). In this research, we propose our own interdependency analysis framework makes up of and interdependency modelling formalism and its object-oriented implementation based on discrete agent-based modelling and simulation and federated simulation. This framework is hereafter referred as Federated Agent-based Modeling and Simulation (Federated-ABMS).

The idea behind Federated-ABMS is the following. A compound complex system, composed of interacting complex systems, can be modelled as a set of interacting agents. Each agent could represent an entire infrastructure, an infrastructure subsystem or humans involved in the scenario. The behaviour of each agent is given by a more or less complex sector specific simulation models, and all the models are integrated using federated simulation. Therefore, the whole model for the compound complex system is a federation of the agent-based models and the sector specific models. Figure 1 shows the Federated-ABMS concept.

The abstraction introduced by Federated-ABMS relieves the modeller of the details of the complex system models (viewed as a black-box), allowing to concentrate her/his attention on the modelling of systems interdependencies. Moreover, Federated-ABMS allow to simulate a system at different level of abstraction: *fast simulation* can be realised when results are needed in a very short time frame, and therefore when is

tolerated a degree of approximation in results, and/or when few data are available to parametrise the model. On the contrary, *computing intensive simulation* is adopted when detailed system model can be parametrise and when there are not strict timelines for results. A typical application of fast simulation is in emergency response, while computing intensive simulation is used in emergency preparedness.

Figure 1   The architecture of the Federated-ABMS framework
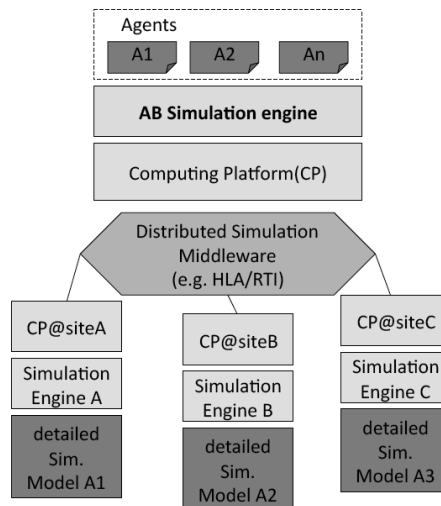
The concept of Federated-ABMS was previously introduced in Casalicchio et al. (2007) where the authors propose a prototype implementation and present preliminary results. In this previous work, the authors did not discuss the agent model and the interdependencies model adopted.

As mentioned before, in the literature there are different research projects that propose agent-based modelling and simulation techniques to study critical interdependent infrastructures, or that aim at integrating existing simulation models to study the behaviour of complex interdependent systems.

This paper contributes to the literature as follow. We formalise the concept of federated agent-based modelling, providing a formalism to model compound complex systems, composed of interacting systems. The introduced formalism allows:

1   to abstract the functional aspects of the infrastructure behaviour, modelled in greater depth reusing existing sector specific models

2   to model direct and cyber interdependencies as service exchange among infrastructures

3   to model geographical and logical interdependencies as model behaviour perturbations.

Furthermore, we outline the steps that characterise the Federated-ABMS methodology, and we show how to apply the methodology to a target system composed of a communication network and of a power grid. Finally, we conclude with a discussion on the implementation of the federated-ABMS framework and related issues.

It is worth to remark that the proposed methodology is not intended as a direct support for decision makers, who need easy-to-use model composition and results visualisation tools. Federated-ABMS is intended as a modelling and simulation methodology for whom want to design and to implement modelling and simulation tools for decision making.

The paper is organised as follows. Section 2 introduces the federated agent-based model formalism. In Section 3, we describe the interdependency model. Section 4 outlines the Federated-ABMS methodology and explain how it can be applied with a case study. Section 5 discusses the Federated-ABMS implementation and related issues. Section 6 presents related works and Section 7 concludes the paper.

2 The federated agent-based model

As shown in many research works, agents are used to model interdependent complex systems (e.g., North and Macal, 2007). A general definition of agent is the following (Bonabeau, 2002):

> "An agent is an entity with a location, capabilities and memory. The entity location defines where it is in a physical space... What the entity can perform is defined by its capabilities... the experience history (for example, overuse or aging) and data defining the entity state represent the entity's memory."

From the above definition emerges that an agent is characterised by its location, behavioural capabilities and memory. A critical infrastructure, also, has is behaviour, it interacts with other systems and with the surrounding environments and humans, and it has (or better its system components has) a location. Moreover, at a give time instant, a critical infrastructure is characterised by an internal state, given by the combination of the internal state of its components. Therefore, a critical infrastructure can be modelled as an autonomous agent and the system composed of interdependent critical infrastructures can be modelled as interacting agents which cooperate and/or compete to realise a common or an individual goal.

An agent a is described by the tuple (V_a, S_a, X_a) where:

1 $V_a = \{v_1^a, \ldots, v_{N_v^a}^a\}$, $v_i^a \in \mathcal{V}_i^a$ and $|V_a| = N_v^a$. V_a is the set of the agent attributes and $\mathcal{V}_i^a$ is the domain of the agent attribute i. The values $v_i^a(t)$ assumed by the agent attributes i at time t represent the state of the agent.

2 $S_a = \{s_1^a, \ldots, s_{N_s^a}^a\}$, $|S_a| = N_s^a$, is the set of services that the agent a provides to other agents. In our model agents interact exchanging services.

3 $X_a = \{x_1^a, \ldots, x_{N_x^a}^a\}$ ($|X_a| = N_x^a$) is the set of inputs of the the agent a. Inputs can be services produced by other agents or perturbations. A perturbation is an unpredictable event that modifies the agent state and alters the behaviour of the agent a, reducing the a's capabilities to provide services. An input is characterised by the tuple $x_i^a = (t_x, x)$ where $x \in \mathcal{X}_i^a$ is the value of the input, $\mathcal{X}_i^a$ the set of possible values for the ith input of agent a, and t_x the time at which the value x is

available ($t_x \in, \mathbb{R}^+$ or $t_x \in, \mathbb{N}^+$ if we consider continuous or discrete time, respectively).

Comparing the proposed federated agent-based model with the definition presented above we have that:

1   the agent state, memory and location are modelled by the agent attributes V_a

2   S_a and X_a model the capability of the agent to interact with other agents providing services and consuming data or services

3   the agent behaviour, that determines how inputs are processed, how services are provided and how the agent state evolves, is modelled using a sector specific model of the complex system modelled.

Figure 2 shows the proposed federated agent model. It is worth to note that only the agent a can interact directly with the detailed model of the complex system abstracted by a.

Figure 2   The federated agent-based model
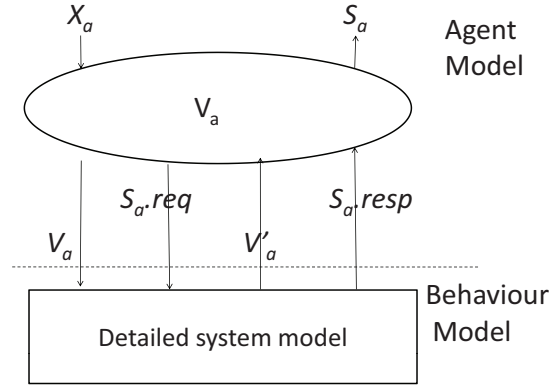
Let us now define the relationship among agent attributes, services and inputs.

The agent state V_a is a function of the time and of the agent inputs X_a, and implicitly of the agent behaviour (as it will be explained in the following). Assuming that the time is discrete, $t \in \mathbb{N}^+$, and that each agent attribute v_i^a depends on a subset of the agent inputs $\{x_{j_1^i}^a, \ldots, x_{j_n^i}^a\}$ we have:

$$v_i^a(t) = f_i^a(t, x_{j_1}^a, \ldots, x_{j_n}^a), \ f_i^a : \mathbb{N}^+ \times \mathcal{X}_{j_1}^a \times \ldots \times \mathcal{X}_{j_n}^a \to \mathcal{V}_i^a.$$

The dependency of the agent attributes on the agent inputs is defined by the mapping[1]:

$$M_a = \{m_{i,j}\}^{N_x^a \times N_v^a}, \ m_{i,j} = \begin{cases} 1 \text{ if } x_i^a \in dom(f_j^a) \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

It is important to remark that f_i^a and M_a depend on the specific system modelled and on the specific goal of the modelling and simulation study, then it is impossible to provide a generic expression for them. In Section 4.1, we give an example of f_i^a and M_a.

The service s_i^a is function of the time, of the agent state, of the agent inputs and of a set of service input parameters $p_1^{i,a}, \ldots, p_{N_p^{i,a}}^{i,a}, p_j^{i,a} \in \mathcal{P}_j^{i,a}$:

$$s_i^a = g_i^a(t, v_{j_1}^a, \ldots, v_{j_n}^a, p_1^{i,a}, \ldots, p_{N_p^{i,a}}^{i,a}) \text{ and } g_i^a : \mathbb{N}^+ \times \mathcal{V} \times \mathcal{P} \rightarrow \mathbb{N}^+ \times [0,1]$$

where $\mathcal{V} = \mathcal{V}_{j_1}^a \times \ldots \times \mathcal{V}_{j_n}^a$ and $\mathcal{P} = \mathcal{P}_1^{i,a} \times \ldots \times \mathcal{P}_{N_p^{i,a}}^{i,a}$.

In our model we assume that $s_i^a = (t, s)$, where $0 \leq s \leq 1$ is the QoS levels of the ith service. $s_i^a = (t, 1)$ if the ith service, invoked at time t', is delivered at time $t \geq t'$ with the 100% of its maximum QoS level. On the contrary, $s_i^a = (t, 0)$ if the service can not be delivered. In the latter case, the time t is meaningless or, depending on the specific service, it can be interpreted as the service time out. The QoS level of a service is described by continuous or discrete a quality funtion.

The last step toward the definition of a federated agent-based model is to provide a solution for:

1 a model of the agent state evolution

2 a model of service delivery

3 a model of service delivery time.

We address issues (1)–(3) using a detailed model of the target complex system. The innovative idea we introduce is to consider the detailed system model as a black-box controlled by the agent model and that computes the new system state, the services delivery time and the service level.

The interaction between the agent model and the detailed system model (see Figure 2) is determined as follow.

1 The agent model requests, at the detailed system model, to compute the new system state V_a' on the basis of the actual agent state V_a and of the services requested $S_a.req$.

2 The detailed system model computes the service response and the service delivery time incapsulates them in the $S_a.resp$ message and reply it to the agent along with the new system state V_a'.

In the proposed solution, the agent model plays the role of the orchestrator of the simulation, while the detailed system model plays the role of a simulation component that receives, from the orchestrator, the system workload $(V_a, S_a.req)$.

3 Interconnecting agents: the interdependencies model

Interdependencies can be classified as (Rinaldi et al., 2001): physical, geographical, cyber and logical. In our model, physical and cyber interdependencies are modelled as service exchange. Moreover, the concept of perturbation allows to model geographical interdependencies. For example, a geographical dependency can be reproduced perturbing in the same way and in the same time frame all the infrastructure located in a given region. While in this work we concentrate our attention on cyber and physical interdependencies, logical interdependencies are out of the scope of our research.

Two agents a and b interact if there is at least one service provided by a that is an input for b: $s_i^a(t) = x_j^b(t)$ for some $1 \leq i \leq N_s^a$ and $1 \leq j \leq N_x^b$. In this case the agent b depends on the behaviour and on the services provided by a, then b and a are dependent. If a depends on b and b on a we have an interdependency and if a and b do not interact directly but interact through a chain of agent interactions we can say that a and b are indirectly interdependent.

To provide a realistic interaction model we consider that an agent a can use a service permanently; for example, a pipe used to transport gas or water, or can be used with a frequency (or probability) $q_{i,j}$, for example a communication line for telecontrol of a power plan or a route for goods transportation. For permanent services $q_{i,j} = 1$, vice-versa $q_{i,j} = 0$ if the service i is not used by the jth input of b.

Then, the interdependencies between agents a and b are modelled by the mappings:

$$Q_{a,b} = \{q_{i,j}\}^{N_s^a \times N_x^b}, \ q_{i,j} \in [0,1] \tag{2}$$

and M_b [previously defined in equation (3)]:

$$M_b = \{m_{i,j}\}^{N_x^b \times N_v^b}, \ m_{i,j} = \begin{cases} 1 \text{ if } x_i^b \in dom(f_j^b) \\ 0 \text{ otherwise} \end{cases} \tag{3}$$

The mapping $Q_{a,b}$ defines how a and b interact, while the mapping M_b defines how b's state is influenced by b's inputs. In the same way, interdependencies can be described by four mappings $Q_{a,b}$, M_b, $Q_{b,a}$ and M_a.

4   The Federated-ABMS methodology

The steps toward the definition of an federated agent-based model are the following:

1   Identification of the simulation study goals.

2   Identification of the infrastructures and systems that characterise the case under study.

3   For each component system identified in Step 2, recognise:

    a   the set of variables that are representative of the system state

    b   the set of services that allow to represent the interaction of the complex system with the other component systems, with the environment and with human beings

    c   the set of perturbations and inputs that influence the component system behaviour

    d   the relationship among agent inputs and agent state variable.

    Steps (a)–(c) should be supported by series of interviews of infrastructures experts.

4   For each system resulting from Step 2, associate an agent a and define the related agent model (V_a, X_a, S_a) and M_a. V_a, X_a S_a and M_a are determined in Steps 3(a)–3(d), respectively.

5   For each couple of agents (a, b) identify the direct interdependencies, for example using interviews of infrastructure experts, and define the interdependencies matrix $Q_{a,b}$.

4.1 The case study

In the following, we apply the federated agent-based methodology to a target complex system composed of an IP communication network (cn) and of a power grid (pg). We suppose that the communication network depends on the power grid, and that there are no auxiliary power mechanisms. For lack of space we concentrate our attention on the above described Steps 4 and 7.

The network state V_{cn} is represented by $\{n_1, \ldots, n_m, l_1, \ldots, l_r\}$ where n_i is a network node (router, access point, switch, ...) and l_j is a network link connecting two network nodes; m is the number of nodes and r the number of links. We assume that $n_i = 1$ ($l_i = 1$) if the node (link) i works and $n_i = 0$ ($l_i = 0$) if the node (link) i does not work.

The agent inputs are $X_{cn} = \{e_1, \ldots, e_m, o_1, \ldots, o_{m+p}\}$ where e_i models the power supply (electricity) for the network node n_i and o_i models an unpredictable system outage for the network node n_i (link l_i). The power supply service e_i used by the communication network is an example of permanent service and the input o_i is an example of probabilistic perturbation. Therefore, o_i is the probability that n_i (l_i) will experiment an outage and it would not work. $e_i = 1$ means that the node n_i is powered by the power grid and $e_i = 0$ means that the node n_i it is not supplied by the power grid. The mapping M_{cn} that models the dependencies of the state variables on the agents inputs is the following:

$$
\begin{array}{c|cc}
& n_1, \ldots, n_m, & l_1, \ldots, l_p \\
\hline
e_1 & & \\
\vdots & I_m & 0_p \\
e_m & & \\
o_1 & & \\
\vdots & & Q_{m+p} \\
o_{m+p} & &
\end{array}
$$

where I_m is an $m \times m$ identity matrix and 0_p is a $p \times p$ null matrix and Q is the outage probability matrix.

In our simplified model the relationship f_{cn} among the agent inputs and agent state is modelled by the following function:

$$
f_{cn} = \begin{cases} n_i = 0 \text{ if } (e_i = 0) \text{ or } ((e_i = 1) \text{ and } (o_i = 1)), \ \forall t \\ l_i = 0 \text{ if } o_i = 1, \ \forall t \\ n_i = 1, l_i = 1 \text{ otherwise, } \forall t \end{cases} \tag{4}
$$

The service provided by the communication network is 'send a message from n_i to n_j' where n_i and n_j are two network access point. Then our simplified network model provides only one service s with two input parameters p_1 and p_2, where p_1 is the source node and p_2 is the destination node. $s = (t_R, 1)$ if the message is delivered at time t_R (the service response time) and $s = (t_{tout}, 0)$ if the service is delivered at $t > t_{tout}$ (t_{tout} is the QoS time out) or if it is not delivered at all because the internal state of the communication network.

To determine the internal state evolution of the communication network on the basis of the agent inputs and service requests we use an event-driven network simulation model implemented using OMNeT++ (http://www.omnetpp.org).

Figure 3 shows the connection between the agent model and the detailed network simulation model.

Figure 3   The federated agent model of a complex interdependent system composed of the power grid (left) and of the communication network (right) (see online version for colours)
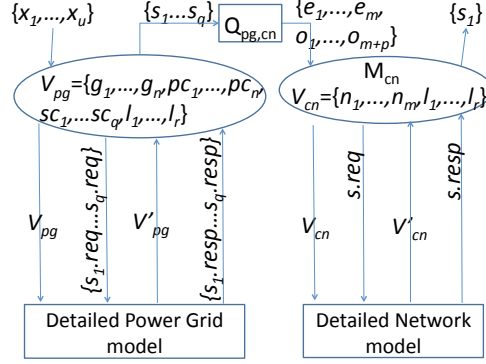
The power grid model considers the following components: power generators (or generation plants) pg, primary cabins pc, secondary cabins sc and distributions/transmission lines d. Then the power grid state is modelled by the set of attributes $V_{pg} = \{pg_1, \ldots, pg_n, pc_1, \ldots, pc_r, sc_1, \ldots, sc_q, d_1, \ldots, d_z\}$ where: $pg_k = 1$ if the generator k work properly and $g_k = 0$ otherwise; $pc_k = 1$ if the primary cabin k work properly and $pc_k = 0$ otherwise; $sc_k = 1$ if the secondary cabin k work properly and $sc_k = 0$ otherwise; and $d_k = 1$ if the distribution or transmission line k work properly and $d_k = 0$ otherwise.

There are many external factors that can influence the power grid behaviour; however, for simplicity, we consider only faults, therefore the power grid agent input is $X_{pg} = \{x_1, \ldots, x_u\}$, $u = n + r + q + z$. If $x_k = (t, 1)$ the power grid component, k will experience a fault at time t. Otherwise, if $x_k = (t, 0)$, the component k does not experiment any outage or it is repaired at time t after a fault at time $t' < t$.

Then we can define $M_{pg} = \{m_{i,j}\}^{u \times u} = I_{u \times u}$ and

$$f_{pg} = \begin{cases} v_i = 1 \text{ if } x_i = 1, \ \forall t \\ v_i = 0 \text{ otherwise}, \ \forall t \end{cases} \tag{5}$$

where v_i is a power grid component pg_i, pc_i, sc_i, d_i.

The service provided by the power grid is 'provide the electricity to the secondary cabin l_k'. We assume that the load is attached directly to the secondary cabins through a bus. Then we have q services: $s_j = (t, 1)$ if the secondary cabin sc_j is operative at time t and $s_j = (\cdot, 0)$ otherwise. The value of s_j depends on the state of all the power grid components (generators, primary cabins and links). In the CRESCO project the power grid behaviour is modelled using a load flow model (that is a time independent model). At time t the power grid simulation model receives as input V_{pg} and it recomputes the power flow, producing the new values for the model state V'_{pg}.

The dependencies between the power grid and the communication network, identified in Step 6, are defined by the mappings:

$$Q_{pg,cn} = \{q_{i,j}\}^{N_s^{pg} \times N_x^{cn}}, \ q_{i,j} = \begin{cases} 1 \text{ if } s_i = e_j \\ 0 \text{ otherwise} \end{cases} \tag{6}$$

and by M_{cn} previously defined.

For simplicity, we do not describe the model of the power grid control functionalities and the dependency of the power grid on the communication network. However, this control rules can be defined or at agent level.

5 Framework implementation and related issues

The implementation of a federated agent-based simulation model is a challenge and there are many open issues. To mention a few: model validation, experiment reproducibility, extensibility to diverse and unforeseen scenarios, simulation scalability, implementation of agents and simulation models federations. In the following, we discuss in detail the last two issues.

5.1 Implementation of agents

In the literature, the problem of discrete agent simulation is widely addressed. There are different frameworks that support agent and multi-agents simulation. Examples are RePast (North et al., 2006), JadeSIM (Gianni, 2008), SIM_AGENT (Sloman and Logan, 1999).

All these approaches have their advantages and disadvantages. Distributed agents (e.g., JadeSIM) allow to design scalable simulation model, some of them are compliant with distributed simulation standard, but they introduce difficulties in designing and testing the simulation logic. Framework such as Repast do not use distributed agents, thus facilitating the design and testing of the simulation logic, but limiting the simulation scalability.

However, Federated ABMS is independent from the technology used to implement agents. Our prototype implementation leverage on RePast as discrete agent simulation framework.

5.2 Federation of the agent-based model(s) and sector specific models

The implementation of the proposed federated agent-based simulation model requires the use of distributed simulation technologies. Distributed simulation allows to integrate together heterogeneous simulation model that can be world wide distributed or locally distributed. Moreover, distributed simulation enable the execution of huge simulations.

If a distributed agents technology is used (see Figure 4) we have, for each infrastructure, a federation composed of the agents model and of the sector specific simulation model (or more then one if needed). The federated agent-based simulation model is obtained federating together all the federations in a unique federation, the *critical interdependent infrastructures federation*.
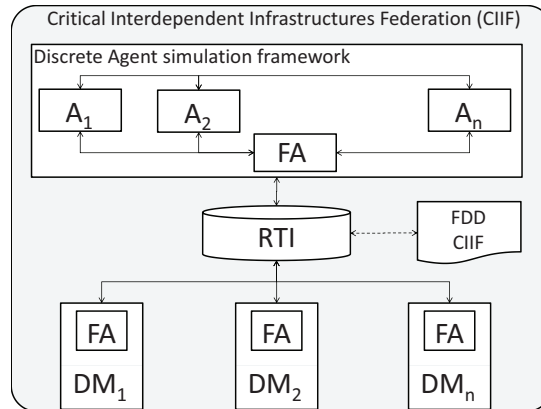
Figure 4   The distributed agents implementation
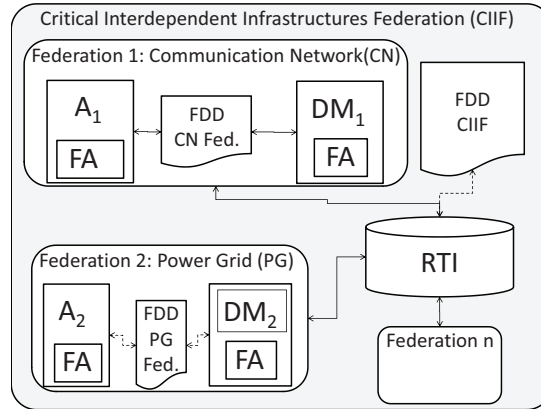
Note: FA is the federate ambassador, A the agent model, DM the sector specific simulation model and FDD the is FOM data document.

If a centralised agent-based simulation framework is used: the framework interacts with all the sector specific simulation models (see Figure 5), while the agents interacts among them using methods invocations. The agent-based simulation framework has a unique federate ambassador, that manages the interaction between each agent and the related sector specific simulation models.

Figure 5   The centralised agents implementation

Note: FA is the Federate Ambassador, A the agent model, DM the sector specific simulation model and FDD is the FOM data document.

Our prototype use the centralised agent approach. We rely on high level architecture (HLA) standard and we used the PoRTIco implementation (The poRTIco Project) of the HLA interfaces. We have modified both the RePast scheduler and the scheduler of OMNeT++ to enable the interaction with PoRTIco. Then, we have implemented the

federate ambassador for both models. The load flow simulator used to model the power grid is a static simulation model, then the integration with PoRTIco was straightforward.

5.3 *Interaction between the agent model and the sector specific model*

The design of the interaction between an agent-based model and the related sector specific model is one of the main challenging problems.

Two aspects have to be considered: the implementation of the physical interaction between models; and the implementation of the logical relationship between the agent state and inputs (V_a and X_a) and the state variable and parameters of the detailed simulation models.

Using the distributed simulation terminology, the physical interaction is defined by the federate object model (FOM). The agent model publishes, as objects, the inputs X_a and the state variables V_a, while the sector specific model publishes V_a' as an object and S_a as an interaction.

The logical relationship is implemented on the agent side. The agent implements the function f_a and the mapping M_a. Each time an agent state variable changes its value, the agent model change the value of the related sector specific model variable. For example, if the network node n_i is a router and $n_i = 0$ at time t, the agent modifies, at time t, the router object published by the OMNeT++ federate.

5.4 *Orchestration of the federated agent-based simulation model*

A distributed application needs an orchestrator process that manages the application logic and the distributed simulation need a process that manages the simulation logic. We named the latter process the simulation orchestrator.

In federated agent-based modelling and simulation, the agent model plays the natural role of the simulation orchestrator. In our implementation the simulation orchestrator is implemented by the RePast *model class* that coordinates the setup and running of the agent model. Each agent is implemented by a Java class, and the *model class* orchestrate the simulation.

On the other hand, if distributed agents are used, a specific agent that works as simulation orchestrator has to be designed.

6 Related works

In this section, we revise the main qualitative and quantitative models proposed to analyse interdependencies in critical infrastructures.

Qualitative analysis approaches are used to identify critical infrastructures, to discover their interdependencies and to generate macro-analysis scenarios.

A simple way to study the failure propagation among interdependent infrastructures is to analyse how the inoperability (i.e., the inability of an element to perform its intended function) is propagated through interconnected infrastructures (Haimes and Jiang, 2001).

A similar, but more general approach, is represented by the influence model (Asavathiratham et al., 2001), where each infrastructure is modelled as a Markov chain, whose evolution is influenced not only by its own state, but also by the states of the chain of the neighbouring sites.

Petri nets (Gursesli and Desrochers, 2003) have been also proposed to study the propagation of failures; however, this approach requires the analyst to consider, besides the places representing the different infrastructures, also a large number of phantom places needed to represent propagated failures.

The difficulties to integrate into a single model all the elements needed to understand the behaviour of critical infrastructures suggested to some authors to adopt a hierarchical holographic modelling (HHM) approach (Ezell et al., 2000; Haimes, 1998). In HHM, the analyst defines a multitude of (mathematical) models, each of them able to clarify and describe some aspects of the problem. Even if each model addresses a different aspect of the systems (components, functions, hierarchical structure, etc.), the overall model offers an acceptable representation of the system.

Another proposal (Svendsen and Wolthusen, 2007a) is to model interaction among infrastructures components and infrastructure users as a direct multigraphs, which can be further augmented by response functions defining interaction between components. In such model vertices of a graph are interpreted as producer and consumer of different type of services. interdependencies are modelled as edges between nodes. This work, which consider only instantaneous interaction, typical of electric power grid and telecommunication sector, is further extended by Svendsen and Wolthusen (2007b) where buffered resources are also modelled. In both cases, the authors analyse network properties, such as robustness, using results from graph theory.

Quantitative approaches to study interdependent critical infrastructure, classified in Donzelli et al. (2006) as system analysis, encompass agent-based modelling (ABM) (Bonabeau, 2002; Rinaldi et al., 2001). ABM uses a bottom-up modelling approach: the whole model is obtained considering a population of agents that autonomously elaborate information and resources in order to produce their outputs. Interactions among agents result in emerging of behaviours that cannot be predicted by any single agent (Rinaldi et al., 2001).

In Dudenhoeffer et al. (2006), the authors describe a framework called CIMS and the work that is being conducted at Idaho National Laboratory to model and simulate infrastructure interdependencies and the complex behaviours that can results. CIMS relies on agent-based modelling, but no details are give on how the agent-based modelling techniques were applied.

In Panzieri et al. (2004), the authors propose a critical infrastructure simulation framework that relies on agents but they do not address the problem of how to model the detailed behaviour of complex infrastructures. In Balducelli et al. (2005), the authors propose an agent-based simulation model of critical infrastructures. They integrate intelligent agents, which behaviour is modelled using fuzzy logic, with a load-flow power grid simulation model. In the paper, there are neither details on the interdependencies model nor on implementation aspects of the simulation framework.

In Cardellini et al. (2007) the authors investigate the adoption of agent-based modelling and simulation to study the interdependencies in critical infrastructures. They adopt UML as modelling language, thus proposing a new way to represent the interdependencies that occur in a complex system composed by different critical infrastructures.

In Casalicchio et al. (2007), the authors propose a methodology which exploits the benefit of both ABMS and federated simulation, to study interdependencies in critical infrastructures. They discus advantages of federated agent-based modelling and difficulties and provide a prototype implementation of a Federated ABMS framework,

which federates an agent-based simulation engine and a simulation environment for information technology systems and communication networks modelling.

SimCip (Beyer et al., 2007) is a simulation framework that relies on agent-based micro simulation and integrate different simulation models. DIESI project (EU Project DIESIS) aims at defining an geographically distributed simulation platform to enable large scale critical infrastructure simulation.

7 Concluding remarks

This paper argued for an alternative agent-based modelling and simulation approach to study interdependent complex systems. The proposed methodology that capitalises the advantages of ABMS and of distributed simulation, is intended as an aid to whom have the challenging task to design a simulation framework for interdependent complex systems analysis.

With Federated-ABMS, a modeller can define an abstract model of the target compound complex system ignoring the details of the component system models (that are used as black-box). This abstraction allows the modeller to concentrate her/his effort in modelling the whole complex system and the system interdependencies. Moreover, the use of distributed simulation allows to build scalable simulation models.

However, the proposed solution has some limitations. First of all, the interdependencies model has to be improved to provide more sophisticated formalism to model geographical and logical interdependencies. Furthermore, model validation mechanisms are not yet investigated.

Acknowledgements

References

Asavathiratham, S., Leiseutre, B. and Verghese, G. (2001) 'The influence model', *IEEE Control System Magazine*.

Balducelli, C., Bologna, S., Di Pietro, A. and Vicoli, G. (2005) 'Analysing interdependencies of critical infrastructures using agent discrete event simulation', *Int. J. Emergency Management*, Vol. 2, No. 4.

Beyer, U., Beyel, C., Cohnitz, C. and Flentge, F. (2007) 'IRRIIS deliverable D2.3.2, IRRISS project', available at http://www.irriss.org.

Bonabeau, E. (2002) 'Agent-based modelling: methods and techniques for simulating human systems', *In Proc. of National Academy of Sciences of the United States of America*.

Cardellini, V., Casalicchio, E. and Galli, E. (2007) 'Agent-based modeling of interdependencies in critical infrastructures through uml', in *ADS '07: Spring Simulation Multiconference 2007/Agent Discrete Simulation 2007*, Norfalk, VA, USA.

Casalicchio, E., Galli, E. and Tucci, S. (2007) 'Federated agent-based modeling and simulation approach to study interdependencies in it critical infrastructures', in *DS-RT '07: Proceedings of the IEEE International Symposium on Distributed Simulation and Real-Time Applications (DS-RT'07)*, Chania, Crete, Greek, IEEE Computer Society.

Donzelli, P., Casalicchio, E., Setola, R. and Tucci, S. (2006) *Modelling and Simulation of Interdependent Critical Infrastructure: The Road Ahead Communication Network and Complex Systems*, Imperial College Press.

Dudenhoeffer, D., Permann, M. and Manic, M. (2006) 'Cims: a framework for infrastructure interdependency modeling and analysis', *Winter Simulation Conference, 2006. WSC 06. Proceedings of the*, 3–6 December, pp.478–485.

EU Project Design of an Interoperable European Federated Simulation Network for Critical InfraStructures (DIESIS), available at http://www.diesis-project.eu/.

Ezell, B., Farr, J. and Wiese, T. (2000) 'Infrastructure risk analysis model', *Int'l Journal of Infrastructure Systems*, Vol. 6, No. 3.

Galli, E. et al. (2009) 'ActivitySim: large-scale agent-based activity generation for infrastructure simulation', *Agent Discrete Simulation Conference 2009*, San Diego, CA, USA, May.

Gianni, D. (2008) 'Bringing discrete event simulation concepts intro multi-agent systems 10th', *Int. Conference on Computer Modeling and Simulation*, IEEE Comp. Soc.

Gianni, D., Loukas, G. and Gelembe, E. (2008) 'A simulation framework for investigation of adaptive behaviors in largely populated building evacuation scenarios', *The International Workshop on Organised Adaptation in Multi-Agent Systems*, AAMAS.

Gursesli, O. and Desrochers, A.A. (2003) 'Modeling infrastructure interdependencies using Petri Nets', *Proc. of Int'l Conf. on Systems, Man and Cybernetics*, October.

Haimes, Y. (1998) *Risk Modelling, Assessment and Management*, Wiley, New York.

Haimes, Y. and Jiang, P. (2001) 'Leontief-based model of risk in complex interconnected infrastructures', *Int'l Journal of Infrastructure Systems*.

North, M., Collier, N. and Vos, J. (2006) 'Experiences creating three implementations of the repast agent modeling toolkit', *ACM Trans. Model. Comput. Simul.*, Vol. 16, No. 1, pp.1–25.

North, M.J. and Macal, C.M. (2007) *Managing Business Complexity: Discovery Strategic Solution with Agent-Based Modeling and Simulation*, Oxford University Press.

Paderson, P., Dudenhoeffer, D., Hartley, S. and Permann, M. (2006) 'Critical infrastructure interdependencies modeling: a survey of US and international research', Idaho National Laboratory.

Panzieri, S., Setola, R. and Ulivi, G. (2004) 'An agent based simulator for critical interdependent infrastructures', *Proc. of Securing Critical Infrastructures Conf.*, October.

Rinaldi, S., Peerenboom, J. and Kelly, T. (2001) 'Identifying, understanding and analyzing critical infrastructure interdependencies', *IEEE Control Systems*, Vol. 21, No. 6, pp.11–25, December.

Sloman, A. and Logan, B. (1999) 'Building cognitively rich agents using SIM Agent toolkit', *Communication of ACM*, Vol. 43, No. 3.

Svendsen, N.K. and Wolthusen, S.D. (2007a) 'Multigraph dependency models for heterogeneous infrastructures', *Critical Infrastructure Protection*, Chapter 24, pp.337–350, Springer.

Svendsen, N.K. and Wolthusen, S.D. (2007b) 'Connectivity models of interdependency in mixed-type critical infrastructure networks', *Inf. Secur. Tech. Rep.*, Vol. 12, No. 1, Elsevier Advanced Technology Publications.

The poRTIco Project, available at http://www.porticoproject.org.

Zhang, P., Peeta, S. and Friesz, T. (2005) 'Dynamic game theoretic model of multilayer infrastructure networks', *Network and Spatial Economics*, Vol. 5, Springer Science.

Notes

1   $dom(f)$ is the domain of the function f and $cod(f)$ the co-domain of the function f.