Analysis and Design of a Multi-Agent System for Simulating a Crisis Response Organization

Rafael A. Gonzalez

Delft University of Technology, Jaffalaan 5,
2628BX Delft, The Netherlands
r.a.gonzalez@tudelft.nl

Abstract. Simulation is a way to deal with the lack of data and difficulty in designing controlled experiments in the field of crisis response. This paper presents the analysis and design of a simulation model used to evaluate different coordination mechanisms for a crisis response organization. Such organizations are often multidisciplinary, short-lived and *ad hoc*. Coordination between the responders can be achieved in a structured way (through standards and hierarchy) or can manifest itself in an adaptive or emergent manner. The characteristics of the response organization and the study of structured vs. emergent coordination fit with the capabilities and nature of multi-agent systems (MAS). The MAS model is built using the GAIA methodology and the JADE agent framework. The model can be configured differently to deal with an emergency scenario developed separately as a discrete-event simulation, providing a testbed for simulating coordination in crisis response.

Keywords: Multi-Agent Systems, Simulation, Coordination, Crisis Response.

1 Introduction

Simulation provides a unique way of understanding complex social phenomena and crisis response organizations in particular [1]. It can be used when the cost of collecting data is prohibitively expensive or there are a large number of conditions to test, as is often the case in crisis response. In situations where large numbers of responders (fire, police, medical, and other agencies) are involved, it is unfeasible to carry out experiments in real-life situations; therefore, simulations offer a valuable platform for testing strategies in advance [2]. Simulation can be used to provide a more economical method of testing contingency plans and practicing coordination between different agencies during crisis response operations [1]. Simulations can illustrate the patterns and pathologies of crisis decision making; they can create a great opportunity for getting acquainted with all aspects of crisis management; and they can help bridge the gap between theory and practice [3]. Simulation is also convenient because it offers a large degree of control for analysts and researchers [1].

This paper presents the analysis and design of a multi-agent system (MAS) to represent a crisis response organization for simulation. The research question that motivates the simulation project is: *How do structured and emergent coordination* *mechanisms between crisis responders perform against each other in terms of effectiveness and efficiency and what are the conditions under which emergent coordination mechanisms perform better?* The simulation approach is both agent-based and discrete-event based. The MAS represents the crisis response organization, which is the subject of the experiments. A discrete-event environment is built alongside to simulate the crisis scenario to which the organization must respond to. The focus of this paper is on the analysis and design of the MAS, while the development of the discrete-event crisis scenario is outside its scope. However, it should be noted that the idea of developing the two dimensions separately is the ability to modify the response organization independently of the crisis scenario used. Conversely, the same MAS organization can be tested with different scenarios built as separate discrete-event simulations. For details on this, see [4].

Before the analysis and design it is worth discussing why a MAS is an appropriate representation of a crisis response organization. When a crisis or emergency occurs it gives rise to an incident organization, which is a temporary organization of otherwise disparate resources drawn from many agencies [5]. Within this incident organization lies a disaster management system comprising the people, technology and procedures concerned with directing resources [5]. Participants in this disaster management system may not have worked together before. Moreover, large-scale emergencies are often beyond the capabilities of the permanent staff and facilities available [6]. The resulting *ad hoc* crisis response teams must be formed quickly, assigned roles and responsibilities, and deployed. The teams are not fixed, but evolve as the availability of personnel, including volunteers, fluctuates. The corresponding entrance and exit of teams increases the difficulty of coordinating the response. As response operations evolve, interactions also need to be redefined for each succeeding situation [7].

Accordingly, a response to an extreme event requires organizational interoperability through a common structure and process, along with the absorption of volunteers and emergent organizations [8]. As a consequence, a crisis requires the reworking of established and standardized procedures through a combination of certain aspects of emergent behaviour and routinized organizational behaviour [9]. Thus, crisis response organizations must be open systems that promote distributed decision making and improvisation in the face of unexpected events or conditions [8].

There is the belief that because the military command and control system is effective in deploying resources, it must be capable of effectively and efficiently providing rescue and relief services, but the military is not trained or structured for the complex tasks of intergovernmental coordination and collaboration needed when preparing for and responding to extreme events [8]. In addition, while hierarchical networks work efficiently during routine operations, they do so poorly in the dynamic environment of emergencies, where node failure may isolate large networks from each other [10]. This has resulted in the tendency towards designing emergent and dynamic networks, rather than formal, static and hierarchical organizations [11, 12]. In practice, most crisis response organizations exhibit some degree of autonomy, while preserving centralization for coordination [8, 11].

In brief, crisis response organizations are fluctuating in size, formed in an *ad hoc* way and multidisciplinary; at the same time, they exhibit hierarchy and centralization together with emergence, autonomy, openness and scalability. If we define an organization as an open system consisting of cognitively restricted, socially situated, and task-oriented actors who interact with other members of the organization and are affected by ambiguity and past experience, then computational models can be used to encapsulate this view and generate predictions regarding the design of an organization for effective performance in response to a crisis [13]. An adequate computational model, given the characteristics of crisis response organizations is a multi-agent system (MAS). Such a system may exhibit similar behaviour, such as a distributed organizational framework, mobility and self-coordination [12, 14].

As a result, MAS are used frequently in crisis response related research. When agents are thought of as functional software units with the capability to execute pre-defined tasks autonomously, they can support the decision-making process of human responders [15]. Agents can extract knowledge from the Internet and inform affected communities and relevant authorities [6, 16]. For example, they can support the decision-making process during the medical response to a large incident, by monitoring news feeds and unloading decision-makers of part of their information-processing needs [17] or by supporting the decisions regarding the distribution of patients in accordance with the availability of resources [18]. They can also be used for fusing the heterogeneous information that they themselves extract [19]. Lastly, decision-support may involve reasoning about mission structures, resource limitations, time considerations, and interactions between teams [20].

Besides decision-support or as a previous step to it, agent-based systems can also be used to simulate the crisis response and its coordination. This role for modelling and simulation has been recognized for decades as a contribution to planning and evaluating response strategies [21]. The value of simulation, as stated in the introduction to this paper, lies in the difficulty or unfeasibility of carrying out experiments in real-life [2]. On one hand, MAS allow designing controlled experiments while at the same time offering the scalability needed for adding (or deleting) roles and rapidly redefining the response organization [7, 21-23]. On the other hand, one of the main uses of agent-based simulation is studying the emergent behaviour of the crisis response organization through the interaction among participating agents [7]. Both capabilities fit well with the nature of a crisis response organization and specifically with coordination as a study objective.

The rest of this paper is structured as follows. Section 2 provides the methodology behind the development of the MAS, first using GAIA [24] and then an implementation-dependent design with GAIA2JADE [25], where JADE is the underlying agent framework [26]. Section 3 shows the results of the analysis phase. Section 4 presents the high-level design (architecture) of the MAS. Section 5 relates to the detailed design and its transition to an implementation-dependent model. Section 6 presents a final discussion and the next steps in this research.

2 Methodology for Developing the MAS

For the analysis and design of our MAS, we have chosen the widely used GAIA methodology [24], which views a MAS as an organized society of individuals in which each agent plays one or more roles and has one or more responsibilities. Each agent interacts with other agents according to a set of protocols and these interactions are seen as the way the agent accomplishes her role in the system.

The GAIA methodology is implementation-independent, which means that it is aimed at analysis and design models. A graphical depiction of the models that should result from following the GAIA methodology is shown in Fig. 1.

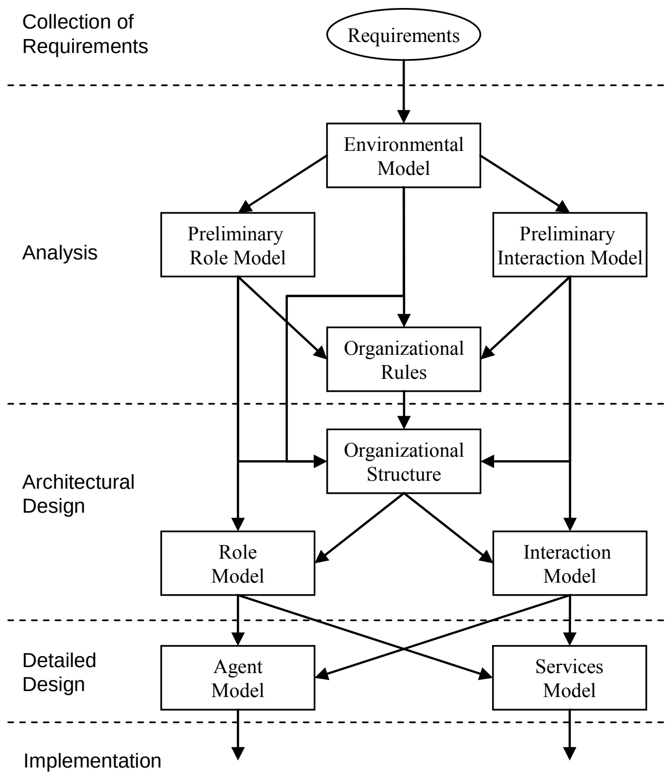

Fig. 1. GAIA Methodology (process and models) adapted from [24]

Because the GAIA methodology is implementation-independent, a transition is expected between the GAIA-based analysis and design of the MAS and an implementation-dependent design. We have chosen JADE [26] as the Java-based agent development framework, due to its widespread adoption, available documentation, open source character and compliance with the FIPA (Foundation for Intelligent Physical Agents) specifications [27]. Also, JADE is one of the frameworks that have already been used for developing MAS in the field of crisis response [17, 22]. We use the GAIA2JADE process [25, 28] as a guide to transform and continue the GAIA method into a JADE-dependent modelling and implementation of the MAS. According to this process, after finishing the GAIA methodology, there are some steps to continue into a JADE-based development, as shown in Table 1.

Table 1. GAIA2JADE Process, adapted from [25].

STEP	INPUT	OUTPUT	COMMENTS
Define communication protocols	GAIA Interactions Model	Domain Ontology; ACL Messages	Messages should comply with FIPA ACL message structure. Sequence diagrams may contribute to modelling.
Define activities refinement table	GAIA Environmental, Interactions, and Roles Models; JADE Domain Ontology	Application Data Class Diagram; Activities Refinement Table	Domain ontology classes are represented as JAVA classes. Algorithms are documented for each liveness property.
Define JADE Behaviours	GAIA Interactions and Roles Models; JADE ACL Messages, Application Data Class Diagram, and Activities Refinement Table	JADE Behaviours Repository	Coding of behaviours in JADE: (1) behaviours are defined; (2) State diagrams are created for each behaviour; (3) constructors are created; (4) behaviour action, input and output are defined; (5) behaviour functionality is added.
Define Jade Agents	GAIA Agent and Service Models; JADE Behaviors	JAVA Code of Agents (in JADE)	All events should be caught in this level.

3 Analysis of the MAS

Before the GAIA based analysis, we went through an initial phase of requirement elicitation, based on identification of response processes for a particular crisis scenario. The processes were extracted from crisis response manuals in The Netherlands [29] and the scenario was adapted from a training case used to describe the Dutch crisis response levels. By using a particular scenario, we were able to limit the number of relevant processes, according to an additional document of guidelines. The result was a list of response procedures and the agencies involved, which served as the basis for identifying the roles for the agents.

The basic response processes are classified according to the responsible discipline, we will focus on this paper in the fire response processes summarized in Table 2.

Table 2. Fire service processes, adapted from [29].

A. SOURCE AND EFFECT CONTAINMENT
Responsible actor: Fire services (regional commander)
1. Fire fighting and containment of dangerous substance emissions
2. Rescue and technical assistance
3. Decontamination of people and animals
4. Decontamination o vehicles and infrastructure
5. Detection (observation) and measurement
6. Warning the population
7. Clearing and providing access

The crisis scenario is a fictitious accident developed for training purposes to illustrate the scaling up of the crisis response as an incident progresses. It goes from a routine response through the four scales of a coordinated response according to the Dutch GRIP (Coordinated Response Procedure) levels. Table 3 describes the scenario from the beginning until it reaches GRIP level 2, because this level is enough to study multidisciplinary coordination without increasing the complexity of the model.

Table 3. Crisis scenario scaling up.

PHASE	DESCRIPTION
Phase 0	The scenario starts with a crane doing work on a road in the jurisdiction of a given municipality.
Phase 1 (Routine)	The incident starts when a truck, carrying flammable liquid, crashes onto the crane. This prompts the response of fire, police and ambulance services in what is initially a routine situation.
Phase 2 (GRIP 1)	Escalation of the incident occurs when the truck catches fire. The incident becomes larger than originally assessed, more response units are needed and a coordinated response is required from multiple disciplines which will setup a CoPI (*Commando Plaats Incident*) operational team, and maintain the mayor of the municipality informed of the situation.
Phase 3 (GRIP 2)	Further escalation occurs when the flammable liquid leaks, the fire spreads and comes into contact with the neighbouring municipality (close to a city). This requires a single leader coordinating the response and that two additional teams be setup, a tactical and a strategic team. The incident is now a regional concern.

3.1 Environmental Model

The environmental model in GAIA is an abstract, computational representation of the environment in which the MAS will be situated. Although GAIA does not provide specific techniques, it can be shown as a list of resources characterized by the type of actions that agents can perform on it [24]. Table 4 shows the resources in the crisis scenario.

Table 4. Crisis scenario resources as a basis for the environmental model.

CRISIS PHASE	RESOURCE	TYPE OF ACTION
0	Road, Obstacle, Housing	Readable
	Vehicles	Changeable
1	Vehicle, Victims, Civilians	Changeable
2-3	Fire	Changeable

It should be noted that the environment itself is later modelled as a discrete-event simulation model, for which these resources would be the entities. Such model is outside the scope of this paper, which focuses on the MAS organization of the crisis response and not on the simulation of the crisis environment.

3.2 Preliminary Role Model

The preliminary role model provides an analysis phase view of the roles and protocols in the MAS, where roles are represented with permissions and responsibilities [24]. Again, we will focus on fire containment due to space considerations. Permissions for the Fireman and OvD (*Officier van Dienst*, Fire Chief) role are presented in Table 5.

Table 5. Roles and Permissions.

ROLE	PERMISSION	RESOURCE
Fireman / OvD	reads	Road, Obstacle, Housing, Vehicle, Civilian
	changes	IncidentVehicle, Victim, Fire

Responsibilities are expressed in terms of liveness properties that describe the state of affairs that an agent must bring about. They are expressed as expressions containing activities (underlined) and protocols (activities that require interaction with other roles – not underlined). Using "x*" means that the activity occurs 0 or more times; "x‖y" means that the activities x and y are interleaved (occur in parallel). Liveness properties of the Fireman and the Fire Officer are shown in Table 6.

Table 6. Liveness properties.

ROLE	LIVENESS PROPERTY
Fireman =	(<u>AssessFire</u>.InformFireAssessment.<u>ContainFire</u>)* ‖ (<u>IdentifyVictims</u>.InformVictimLocation)*
OvD =	(AnalyseFireSituation.PlanContainment.<u>CommunicateContainmentPlan</u>. <u>GetContainmentResources</u>.DeployContainment.<u>SuperviseContainment</u>)*

3.3 Preliminary Interaction Model

The preliminary interaction model captures the dependencies and relationships between the various roles in the MAS organization [24]. Each interaction protocol is defined in terms of: name, initiator, partner, inputs and outputs. The protocols for the liveness properties in Table 6 are shown in Table 7.

Table 7. Preliminary Interaction Model.

PROTOCOL NAME	INITIATOR	PARTNER	INPUT	OUTPUT
InformFireAssessment	Fireman	OvD	Site assessment	Message to commander
InformVictimLocation	Fireman	OvD	Site assessment	Message to commander
AnalyseFireSituation	OvD	Officers	Nature, scope and expected evolution	Fire analysis
PlanContainment	OvD	Officers	Fire analysis	Containment plan
DeployContainment	OvD	Fireman	Containment plan, resources received	Resources deployed

4 Architectural Design of the MAS

A MAS architecture is equivalent to its organizational structure, which in turn is a result of combining the system topology and the control regime [24].

4.1 Organizational Structure

A topology for the MAS organizational structure may be peer-to-peer, hierarchical, multi-level or composite. Given the initial discussion of a crisis response organization, topology in this case needs to combine the hierarchy explicitly designed into the response disciplines, with the lateral relationships possible between first responders and commanders. The control regime can be based on specialization or partition. In a crisis response organization (homogeneous) partitioning occurs within disciplines and (heterogeneous) specialization occurs in between disciplines. The resulting structure is depicted semi-formally in Fig. 2. Besides the role of Fireman and OvD shown above, this structure also includes equivalent structures of other roles not shown due to space considerations: for the police (Policeman and Police Chief, OvD-P), medical services (medics and Medical Officer, OvD-G) as well as regional commanders for each discipline (CvD, *Commander van Dienst*) and an overall Operational Leader (OL).

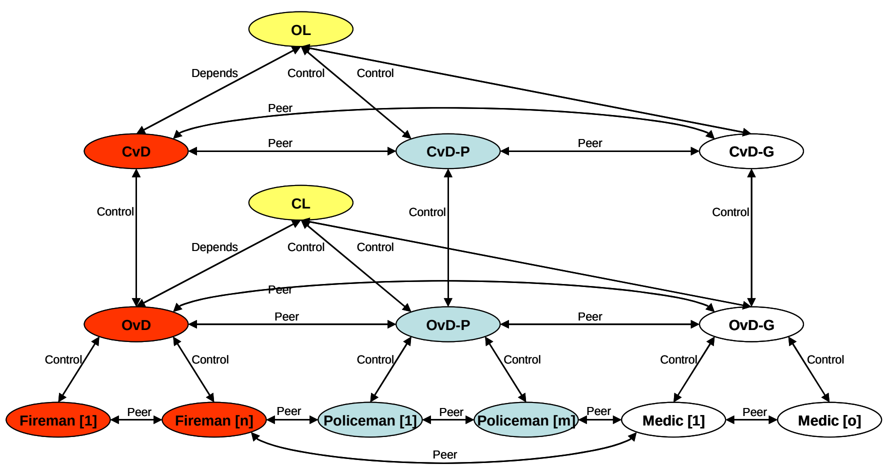

Fig. 2. Organizational Structure of the MAS

4.2 Role Model

After having defined the organizational structure, the preliminary role model can be revised, resulting in a detailed role model for each of the final roles. To illustrate a role model with one example, Table 8 shows the Fireman role.

Table 8. Fireman Role.

ROLE	Fireman
DESCRIPTION	The Fireman is the Fire Services field agent in charge of fighting (suppressing) fires and rescuing victims.
PROTOCOLS & ACTIVITIES	GetToLocation, NotifyArrival, AssessSituation, InformAssessment, UpdateAssessment, InformResult, ContainFire, MoveVictim
PERMISSIONS	Read Civilian, House, Vehicle. Change Fire, Responder (proxy simulated fireman: self)
RESPONSIBILITIES	Fireman = GetToLocation. NotifyArrival.(AssessSituation. InformAssessment.(Respond. UpdateAssessment. InformResult)*)* Respond = ContainFire | MoveVictim

4.3 Interaction Model

The interactions model represents the interaction between the agents, connected through input/output. The fire containment protocols are shown in Fig. 3. Although the model is sequential, interactions are shown between initiating (left of each box) and receiving agents (to the right). Dotted arrows represent conditional transitions. After Communicate Plan, other actions continue, but are left out for lack of space.

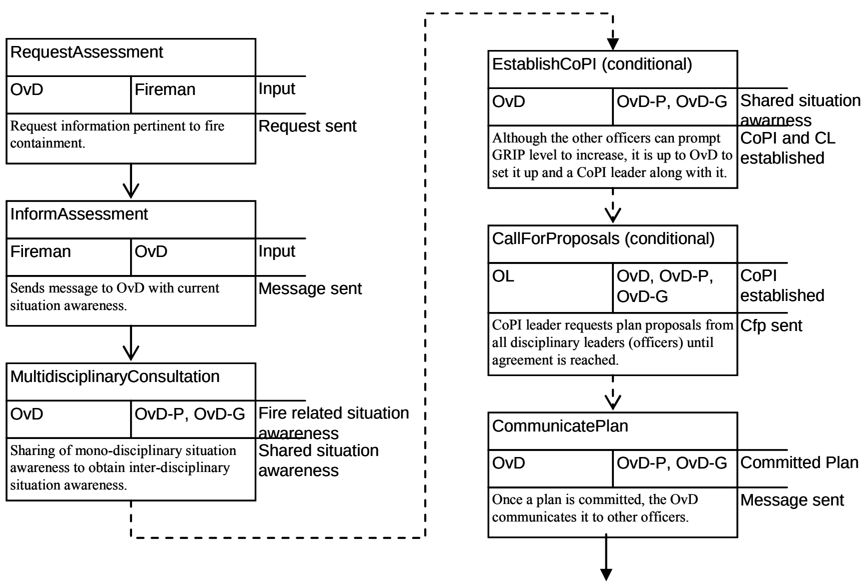

Fig. 3. Interaction Model for Fire Containment

5 Detailed Design

This section contains the detailed design of the agent-based aspects in the form of an agent and a services model.

5.1 Agent Model

The agent model defines the agent classes that will play specific roles [24]. In our case, the OL role is absorbed by the CvD agent (in practice this is what usually occurs in an emergency). Similarly, the CL role is absorbed by the OvD. All commanders and officers will have only one instance. Following the notation suggested in [28], the agent model is presented in Fig. 4 where blocks represents agent types, rounded figures represent roles and "*" means 0 or more instances.

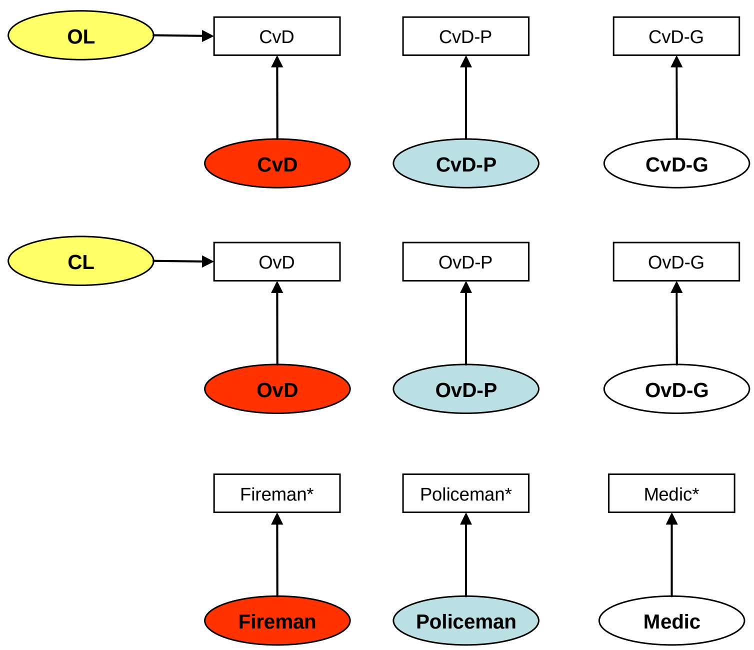

Fig. 4. Agent Model

5.2 Communication Protocols

The first implementation-dependent step that follows the GAIA2JADE process [25] is defining the communication protocols for the agents, through an ontology and a set of ACL messages. The domain ontology in Jade describes the elements that agent use to create the content of messages, specifically concepts, predicates and actions [26]. Concepts are the semantic elements of the vocabulary. Predicates are the structural elements. Actions are special concepts that denote agent actions. Table 9 describes the domain ontology (omitting the attributes).

Table 9. Domain Ontology.

ELEMENT	NAME	DESCRIPTION
Concept	Civilian	Civilian (victims or not)
Concept	Estimated Population	Population observed by a responder
Concept	Fire	Observed fire
Concept	Infrastructure Element	Housing, object, vehicle or obstacle
Concept	Location	Defined location
Concept	Resource	Material response resource
Concept	Responder	Information about a responder
Concept	Strategy	Response strategy
Concept	Time	Timestamp of observations
Concept	Traffic	Perceived traffic
Predicate	Situation Assessment	Current observation of the incident
Action	Alarm	Alarm message
Action	Plan	Response plan per discipline

Given the interaction protocols defined in the GAIA design and the above ontology, ACL messages according to FIPA [27] can be defined. As an example, Table 10 presents the interaction protocol for *RequestAssessment* as a FIPA Query.

Table 10. RequestAssessment FIPA Query.

ACL MESSAGES	SENDER	RECEIVER	FIPA PERFORMATIVE
Request Assessment	Officer	Responder	Query-ref
Query Not Understood	Responder	Officer	Not understood
Refuse Query	Responder	Officer	Refuse
Query Failure	Responder	Officer	Failure
Inform Assessment	Responder	Officer	Inform

5.3 Activities Refinement Table

This step defines the activities refinement table, where application-dependent data, their structure and the algorithms that are going to be used by the agents are defined [25]. The table is meant to specify the liveness properties of the agents, having defined the ontology. Under read and change, there is a reference to data classes (no longer environmental objects, but ontology-dependent classes). Under Description there is a top-level algorithm in pseudocode for the corresponding activity. As an example, Table 11 shows the activity refinement table portion for the Fireman role.

Table 11. Activity refinement table for Fireman role.

ROLE	ACTIVITY	READ	CHANGE	DESCRIPTION

Fireman	Fireman	Resource
		Weather
		Responder
		Location
		Element
		Fire
		Civilian
			-	**do** GetToLocation
				do NotifyArrival
				while Strategy.exit != [exit criteria]
				  do AssessSituation
				  do InformAssessment
				  while fire != null || civlians.status != victim
				    do Respond
				    do UpdateAssessment
				    do InformResult
				  end while
				end while

Fireman	Respond		Fire
			Civilian
				if assigned Fire
				  do ContainFire
				else if assigned Victim
				  do MoveVictim
				end if

5.4 Jade Behaviors

This step implements Gaia activities as Jade Behaviours [25]. First, behaviours are defined. Second, a state diagram (UML) is provided for each relevant behaviour to help identify data exchange between behaviours and easily map to Jade FSM (Finite State Machine) behaviours. Jade behaviours are defined from Gaia activities, through mapping activities. All Gaia liveness formulas are translated to JADE behaviours. In this case we use the one for Fireman defined in the Responsibilities section of **Table 8**. As a general rule, the "•" operator in a liveness formula denotes that the behaviour at the left-hand side is complex, while the [], +, *, | operators denote that the left-hand side can be a finite state machine. All behaviours should inherit from the *jade.core.behaviours.Behaviour* class. As an example, we provide the FSM diagram in UML for the Fireman agent in Fig. 5. Implementation follows from the bottom-up (from simple to complex behaviours). This results in one FSM diagram for each agent which is subsequently implemented as FSM and FSM Child Behaviors in JADE.

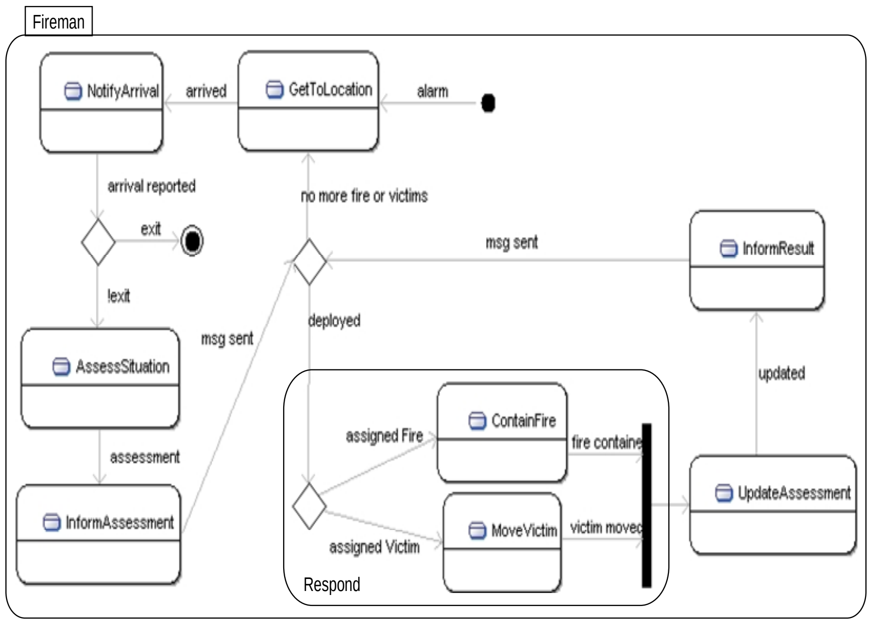

Fig. 5. FSM diagram for Fireman agent.

6 Conclusion

This paper summarized the processes of analysis and design of a multi-agent system for a crisis response organization with the purpose of building a simulation testbed to experiment with different coordination mechanisms. With respect to the use of the GAIA methodology, it proved to be a structured way of performing analysis and design. In this case, Organizational Rules and a Service Model were not needed, due to the fact that the agent services are not meant for "consumption" but rather for simulation. They could be specified in the future so the same agent behaviour could be used not for simulating crisis response agents, but rather to support real crisis response agents with information processing tasks.

The transition from a GAIA-based analysis and design to a JADE-dependent design proved to be relatively straightforward through the use of finite state machines and corresponding JADE behaviours. Indeed, the FSM representation for the agents was chosen due to the facilities offered by JADE, but it could be seen also a source for formal models of the agents that could directly be simulated. In addition, having expressed the interaction protocols with ACL messages enables implementation of agent communication in accordance with FIPA specifications. This allows reusing and complying with a predefined set of interaction protocols.

Running simulation experiments with this model will permit comparing between coordination strategies in terms of their effectiveness (damage reduction and protection of civilians) and efficiency (performance and response time). By running the agents repeatedly over the same scenario, making variations over the coordination mechanisms as expressed through interaction protocols, through centralized vs. peer to peer communication and through standards (embedded in the FSM structure) vs. emergence (occurring when agents behave autonomously), the simulation can be used to experiment and analyse different configurations that can be used to inform coordination theory in crisis response or as basis for developing ICT services that support responders in the field.

After the experiments and validation, this research will also be able to show results with respect to the integration of MAS and discrete-event simulation in a single model, the (dis)advantages of JADE for simulation purposes, and the rigidity that may arise from using GAIA: we consider these to be interesting points for future research.

References

1. Kleiboer, M.: Simulation Methodology for Crisis Management Support. Journal of Contingencies and Crisis Management 5 (1997) 198-206
2. Suárez, S., Collins, J., López, B.: Improving Rescue Operations in Disasters: Approaches about Task Allocation and re-scheduling. 24rd Annual Workshop of the UK Planning and Scheduling Special Interest Group (PlanSIG), London (2005) 66-75
3. Boin, A., Kofman-Bos, C., Overdijk, W.: Crisis Simulations: Exploring Tomorrow's Vulnerabilities and Threats. Simulation Gaming 35 (2004) 378-393
4. Gonzalez, R.A.: Crisis response simulation combining discrete-event and agent-based modeling. In: J. Landgren, Nulden, U., Walle, B.V.d. (eds.): 6th International ISCRAM Conference (ISCRAM2009), Gothenburg, Sweden (2009)
5. Smith, W., Dowell, J.: A case study of co-ordinative decision-making in disaster management. Ergonomics 43 (2000) 1153-1166
6. Liu, K.F.R.: Agent-based resource discovery architecture for environmental emergency management. Expert Systems with Applications 27 (2004) 77-95
7. Comfort, L.K., Ko, K., Zagorecki, A.: Coordination in rapidly evolving disaster response systems: The role of information. American Behavioral Scientist 48 (2004) 295-313
8. Harrald, J.R.: Agility and Discipline: Critical Success Factors for Disaster Response. The ANNALS of the American Academy of Political and Social Science 604 (2006) 256-272
9. Dynes, R.R., Aguirre, B.E.: Organizational adaptation to crises: Mechanisms of coordination and structural change. Disasters 3 (1979) 71-74
10. Kapucu, N.: Interagency communication networks during emergencies: Boundary spanners in multiagency coordination. American Review of Public Administration 36 (2006) 207-225
11. Houghton, R., Baber, C., McMaster, R., Stanton, N., Salmon, P., Stewart, R., Walker, G.: Command and control in emergency services operations: a social network analysis. Ergonomics 49 (2006) 1204-1225
12. Weigand, H.: Agent Community Support for Crisis-Response Organizations. In: Meersman, R., Tari, Z., Herrero, P. (eds.): On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops. Springer, Berlin (2006) 218-226
13. Lin, Z., Zhao, X., Ismail, K.M., Carley, K.M.: Organizational Design and Restructuring in Response to Crises: Lessons from Computational Modeling and Real-World Cases. Organization Science 17 (2006) 598-618
14. Dooley, K., Corman, S.: Agent-based, genetic, and emergent computational models of complex systems. In: Kiel, L.D. (ed.): Encyclopedia of Life Support Systems (EOLSS). Unesco / Eolss Publishers, Oxford (2002)

15. Gadomski, A.M., Bologna, S., di Costanzo, G., Perini, A., Schaerf, M.: Towards intelligent decision support systems for emergency managers: the IDA approach. International Journal of Risk Assessment and Management 2 (2001) 224-242
16. Goh, O.S., Fung, C.C.: Automated Knowledge Extraction from Internet for a Crisis Communication Portal. In: Wang, L., Jin, Y. (eds.): Fuzzy Systems and Knowledge Discovery: Second International Conference, FSKD 2005, Changsha, China, August 27-29, 2005, Proceedings, Part II Springer, Berlin (2005) 1226-1235
17. Bloodsworth, P., Greenwood, S.: COSMOA an ontology-centric multi-agent system for co-ordinating medical responses to large-scale disasters. AI Communications 18 (2005) 229-240
18. Zhu, S., Abraham, J., Paul, S., Reddy, M., Yen, J., Pfaff, M., DeFlitch, C.: R-CAST-MED: Applying Intelligent Agents to Support Emergency Medical Decision-Making Teams. In: Bellazzi, R., Abu-Hanna, A., Hunter, J. (eds.): Artificial Intelligence in Medicine. Springer, Berlin (2007) 24-33
19. Pavlin, G., de Oude, P., Nunnink, J.: A MAS Approach to Fusion of Heterogeneous Information. 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05). IEEE Computer Society (2005)
20. Wagner, T., Phelps, J., Guralnik, V., VanRiper, R.: Key-Based Coordination Strategies: Scalability Issues. In: Scerri, P., Vincent, R., Mailler, R. (eds.): Coordination of Large-Scale Multiagent Systems. Springer, New York (2006) 147-171
21. Jain, S., McLean, C.: Simulation for emergency response: a framework for modeling and simulation for emergency response. In: Chick, S., Sánchez, P.J., Ferrin, D., Morrice, D.J. (eds.): Proceedings of the 2003 Winter Simulation Conference. Winter Simulation Conference, New Orleans, Louisiana (2003) 1068-1076
22. Massaguer, D., Balasubramanian, V., Mehrotra, S., Venkatasubramanian, N.: Multi-Agent Simulation of Disaster Response. In: Jennings, N.R., Tambe, M., Ishida, T., Ramchurn, S.D. (eds.): First International AAMAS Workshop on Agent Technology for Disaster Management. ACM, Hakodate, Hokkaido, Japan (2006) 124-130
23. Robinson, C.D., Brown, D.E.: First responder information flow simulation: a tool for technology assessment. In: Brown, D.E. (ed.): 37th Conference on Winter Simulation. IEEE, Orlando, Florida (2005) 919-925
24. Zambonelli, F., Jennings, N.R., Wooldridge, M.: Developing multiagent systems: The Gaia methodology. ACM Transactions on Software Engineering and Methodology (TOSEM) 12 (2003) 317-370
25. Moraitis, P., Spanoudakis, N.: The GAIA2JADE Process for Multi-Agent Systems Development. Applied Artificial Intelligence 20 (2006) 251 - 273
26. Bellifemine, F., Caire, G., Trucco, T., Rimassa, G.: Jade Programmer's Guide: JADE 3.6. http://jade.tilab.com/doc/programmersguide.pdf (2008)
27. IEEE Foundation for Intelligent Physical Agents: The Foundation for Intelligent Physical Agents. http://www.fipa.org/index.html (2009)
28. Moraïtis, P., Petraki, E., Spanoudakis, N.: Engineering JADE Agents with the Gaia Methodology. In: Kowalczyk, R., Müller, J.P., Tianfield, H., Unland, R. (eds.): Agent Technologies, Infrastructures, Tools, and Applications for E-Services. Springer, Berlin (2003) 77-91
29. Ministerie van Binnenlandse Zaken en Koninkrijksrelaties: Handboek Voorbereiding Rampenbestrijding. Deel B. Hoofstuk 3. Rambbestrijdingsprocessen. http://www.handboekrampenbestrijding.nl/contents/pages/3794/b-3.pdf (2003)