A. Oulhaci, E. Tranvouez, S. Fournier, B. Espinasse, « A Multi-Agent Architecture for collaborative Serious Game applied to Crisis Management training: improving adaptability of Non Played Characters », The 7th European Conference on Games Based Learning, ECGBL 2013, Porto, Portugal, October 3-4, 2013.

A MultiAgent Architecture for collaborative Serious Game applied to Crisis Management training: improving adaptability of Non Played Characters

M'hammed Ali Oulhaci[1,2], Erwan Tranvouez [1], Sébastien Fournier[1], Bernard Espinasse[1] [1]
Aix-Marseille Université (AMU), CNRS, LSIS UMR 7296, Marseille, FRANCE
[2] Groupe SII, Société d'Ingénieur et de Conseil en Technologies, Aix-en-Provence, FRANCE

{ali.oulhaci, erwan.tranvouez, sebastien.fournier and bernard.espinasse}@lsis.org

Abstract: Serious Games (SG) are more and more used for training in various domains, and notably in the crisis management domain. Crisis management can gather several hundred stakeholders, which can present various difficulties when organizing field exercises. Serious Game constitutes a more practical alternative with specific benefits concerning detailed players' actions tracking during a simulated exercise. Moreover, Non Played Characters (NPC) can be used to adapt the crisis management exercise perimeter to the available stakeholders as well as specific training objectives. In this paper we present a general Multi-Agent System (MAS) architecture providing support to the behavioral simulation as well as the monitoring and assessment of human players. To each NPC is associated a so called Game Agent designed to reproduce the behavior of the actor simulated. The Game Agents are based on a deliberative model (Belief Desire Intention) with added editing features to facilitate the scenario design phase. Thus an Agent editor allows a designer to configure agents' behaviors illustrated in this paper with the case of crisis management scenario. The behavior simulation was implemented within the preexisting SIMFOR project, a serious game for training in crisis management.

Keywords : Serious Game, Multi-agents system, Multi-agent Simulation, Crisis Management.

1. Introduction

Serious Games (SG) are more and more used for training in various domains, and notably in the crisis management domain. Crisis management can gather several hundred stakeholders, which can present various difficulties when organizing field exercises. Serious Game constitutes a more practical alter-native with specific benefits concerning detailed players' actions tracking during a simulated exercise. Moreover, Non Played Characters (NPC) can be used to adapt the crisis management exercise perimeter to the available stakeholders as well as specific training objectives. The work presented in this paper focuses on adding NPC capabilities to the preexisting SIMFOR Serious Game dedicated to training actors (with various level of expertise) involved in a Crisis Management situation. NPC capabilities implies being able to simulate humans' behaviors with whom human players interact.

This paper addresses the modeling and software requirements needed to support these objectives. A general Multi-Agent System (MAS) architecture has thus been proposed providing support to the behavioral simulation as well as the monitoring and assessment of human players. To each NPC is associated a so called Game Agent designed to reproduce the behavior of the actor simulated (role incarnated). The Game Agents are based on a deliberative model (Belief Desire Intention) which is quite usual in Multi-Agent Systems for complex behavior modeling (and simulation). This feature can be summarized in describing a agent as pursuing multiple goals (possibly with different priorities), which can be attained by plans composed of a sequence of actions either applied in the virtual world (3D environment) or resulting in interactions between agents/human players (with a self-evolutionary response). To facilitate the scenario design, we have implemented an agent editor which will allow a designer to configure agents' behaviors (as well as dialogues) for a SG scenario, applied here to Crisis Management training.

The next section presents the SIMFOR project, a serious game for training crisis management. Section 3 discuss the NPC issue and relates to different works in the field of multi-agent systems and behaviors simulation. In section 4 we define our Game Agent model for the SIMFOR project and in section 5, we present a short game scenario in the SIMFOR project to illustrate how the implementation phase is supported. Finally we conclude and present future works for the SIMFOR project.

2. The SIMFOR project

In this section, we briefly present SIMFOR project, a serious game for crisis management as well as the the general architecture combining Intelligent Tutoring System (ITS) and Serious Game (SG) elements.

2.1 SIMFOR context

SIMFOR (figure 1) is a serious game developed by SII [1] company in partnership with Pixxim [2] company, in response to serious gaming call for project launched by the French Secretary of State for Forward Planning and Development of the digital economy. SIMFOR provides a fun and original approach for learning crisis management as a serious game. SIMFOR is adapted to actors' needs and enables learners to train for major crisis management by integrating multi-stakeholder aspect (ie heterogeneous learning profiles). The project objective is to create a training environment that immerse users in a crisis management situation in real-time context and realistic in terms of environment, self-evolving scenarios and actors.

Figure 1: Screenshot from SIMFOR project

SIMFOR is a multi-player game and allows different people to learn skill (shared or specific) in the same game. This is possible because SIMFOR does not target only the specialists in the field of crisis management, but rather the non-professional. Managing a major crisis can mobilize several hundred stakeholders, from the regional Prefect in his office to the firefighter in the field. These stakeholders are required to communicate and work together in order to restore a normal situation.

2.2 The general architecture of the system

The SIMFOR architecture combines elements from the Intelligent Tutoring System and Serious Game domains (see (Oulhaci et al., 2013a) for a detailed presentation). Our goal is to associate the playful learning of SG and the different modules of an ITS (domain model, learner model, pedagogical model) to get the optimal learning environment. The SIMFOR architecture is composed of the following components:

- **The SG module (SIMFOR):** this module includes the 3D models, user interface (as a communication channel between the learner and the system), simulation module (for natural phenomena such as fire propagation), and data models. This module constitutes the former "perimeter" of the Simfor SG.
- **The Behaviors Simulation module:** which allows simulating humans behaviors to replace absent players with "artificial" actors (Game Agent).

1. http ://www.groupe-sii.com
2. http ://www.pixxim.fr

The Evaluation module: the evaluation module aims to provide skills assessment of players in real time to the pedagogical module.

The Pedagogical module: which plays the role of a virtual tutor accompanying the learners by providing support and help during (and after) their training.

Knowledge representation module: All knowledge used or produced by the previous modules of our proposed architecture is stored in the following models:

– **The Domain model:** the domain model represents the different concepts of crisis management and it's segmented into parts representing a role or a skill to learn.

– **The learner model:** for each learner or agent, a learner model is associated. This model represents the mental state of actors at a time t.

As this paper focuses on simulation of human actors in a SG, the following section exposes the scientific issue of NPCs in the SIMFOR project and some relevant work in this field.

3. Adaptive NPC for SIMFOR

The SIMFOR project faces two issues:

- The simulation of human behavior of NPC players.
- The monitoring and evaluation of learners during their training.

The learner assessment was discussed in (Oulhaci et al., 2013b) and (Oulhaci et al., 2013a). To deal the heterogeneous aspect of the learner assessment (assess different skills and trades), we have pro-posed the concept of the "Evaluation Space". The guiding idea is to consider a SG scenario through different view, each corresponding to a particular evaluation objective. An "Evaluation Space" thus gathers (homogeneous) information and primitives to manipulate these information in order to produce assessments, such as a Behavioral Space (for evaluating procedural knowledge) or Social Space (for evaluating actors interaction during a game scenario).

Adding NPC capabilities to the SIMFOR SG implies being able to simulate actors' behaviors with whom human players interact (in the best case scenario without knowing the virtual nature or not of other players). This step requires extracting from domain experts nominal behaviors which players are expected to follow, and express them in suitable format. The challenge of behaviors simulation is how to transform an expert domain nominal behavior to a SG NPC behavior? Given the number of stakeholders' as well as their skill heterogeneity, designing a scenario to establish crisis management exercises is a complex task. We present a scenario example in section 3.1 to discuss this issue.

3.1 NPC and complex behavior simulation

As a SG, SIMFOR aims at immersing players in a virtual world enabling them to pretend acting as they would (and should) do in a real emergency situation. Knowledge and skills involved in such situation are various in nature as well as in terms of evaluation means, but nonetheless must be all assessed in order to certify (or not) that players know their part of the job on which many lives may depend. To better understand the heterogeneous aspect of the behaviors simulation as well as the assessment needs, let's consider a simplified example of emerging situation scenario.

This scenario starts with a TDM (Transport of Dangerous Material) truck overturned after a traffic accident. The tank is damaged and hydrocarbon is spreading over the road. A witness to the accident gives the alarm by calling the CODIS (Departmental Center for Operational Fire and Rescue Services in French) which in turn must perform four missions consequently to the alert. First, CODIS has to send a Firefighter on the scene to retrieve information about the accident ("send firefighter"). Once information on the accident is received (transmitted by the Firefighter in the ground), confirming a TDM accident has occurred, the CODIS must secondly gives instructions to an Officer (firefighter) on the measures to be taken. In a TDM accident the Officer must give the intervention order (send another Firefighter with a fire truck). Then thirdly, the CODIS must complete an information sheet on the disaster that passes later through a fax to the Mayor, Prefect and the Sub-Prefect (sending order is not important). Finally the last mission is to inform the OCP officer (Operational Command Post) once it is sent by the Prefect.

This scenario excerpt illustrates the needs of the domain, for behaviors simulation (define nominal agent behavior), as well as for the assessment process (if the CODIS is played by a human). Actors can play the same role (Firefighter), but enact different behaviors (one collect disaster information, and the other should intervene on the disaster). Moreover, we have a trades which do not belong to a sole organization (given the large number of stakeholders in crisis management), like School Principal, Mayor, media, etc. In addition, the actors' behaviors may differ from a scenario to another (depending of the disaster nature ie fire, earthquake...). Therefore, the scenario designer (domain expert) must specify the actors involved their associated behaviors, disasters consequences, etc, for each scenario exercise. In the next section we present some works in the field of behavior simulation and multi-agents architecture and how our work relates to these works.

3.2 Related work

In this section, we present some representative works related to behaviors simulation in Serious Games (SG). The Artificial Intelligence (AI) has always been present in video games, more or less elaborated (depending on the video game objective) (Bakkes et al., 2009). SGs borrow much from classic video games, but SGs are more than game, a SG is a game that is used to learn something. The tools and mechanisms borrowed to the video game must be more elaborated, like AI. The AI is used to simulate natural phenomena or human behaviors to get a realistic virtual word for training (Zyda, 2005). In (Buche et al., 2003), Buche propose MASCARET, a pedagogical multi-agent system for virtual environment for training. The MASCARET model is proposed to organize the interactions between agents and to endow them with reactive, cognitive and social abilities to simulate the physical and social environment. The physical environment represents, in a realistic way, natural phenomena. The social environment is simulated by agents executing collaborative and adaptive tasks. The MASCARET model was applied to SECUREVI, an application for fire-fighters training. The MASCARET model allows designing a complex organization (by defining role, organization, behaviors ...), but is difficult to transpose in situations where many and heterogeneous organizations interact. For each organization, we must define roles and procedures etc, which are not adapted for larger scale crisis management exercises. In addition, the SECUREVI project does not include any assessment solution.

More specifically on multi-agent organizational modeling, the MOISE+ model (Hubner¨ et al., 2002) (Model of Organization for multi-agent SystEms) considers organizational structure and dynamics of a Multi-Agent System or MAS (for example for simulation purpose). This model adds an explicit deontic relation (to structure and dynamics) to make the (artificial) agents able to reason on the fulfillment of their obligations or not. In these models, obligations and permissions are entitled to roles (such as a firefighter has to extinguish a fire and may use a fire hose which a bystander may not). As training may require actors to detect wrong behaviors (during a collaborative task between a simulated actor and human player), we would also need to allow (voluntary) erroneous behaviors which is not covered by MOISE+.

To simulate NPCs in the SIMFOR project, we have opted for BDI architecture. The BDI (Beliefs, Desires, and Intentions) model is an agent modeling standard in the field of agent behavior modeling, inspired from the human reasoning process (Rao et al., 1995), and has been widely applied. A BDI model is based on the notion of capacity, skills, beliefs, purpose, desire, and intention-plan. Agents aim at achieving their goals by executing plans depending of their current knowledge (beliefs). These concepts allow designing and programming agents with complex behaviors. Our goal in the SIMFOR project is to provide an agent architecture helping the domain expert to design NPCs for a specific scenario. These NPCs can have a nominal behaviors (perform the expected behaviors) or intentionally erroneous behaviors. The NPCs must also adapt their behaviors related to other players (learners) actions and interaction (social abilities) as well as events from the environment (3D world) that can occur during the game.

4. A BDI architecture for SIMFOR

The proposed model is composed of a set of agent, actions and facts. Each SIMFOR scenario is associated with several agents model that reflect the NPCs behaviors. An agent model is represented as follows:

$$Model_{(role)} = \{Goals; Plans; Facts; Dialogues\} \quad (1)$$

A Game Agent will play a role in a scenario, and as such tries to achieve Goals (activated ie evaluated as reachable, by the context) by enacting its associated plans depending on its knowledge of the situa-

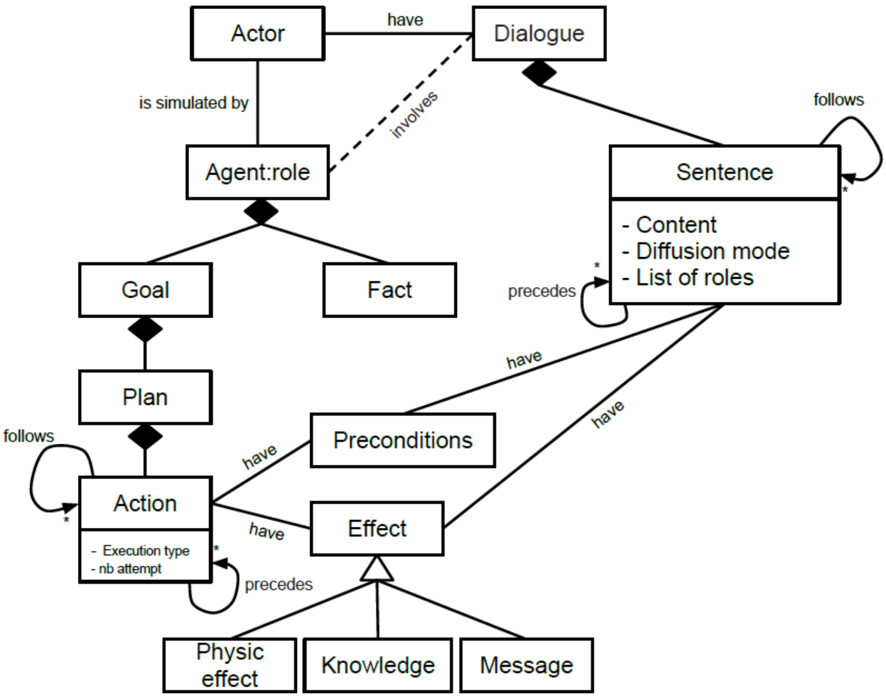

Figure 2: The SIMFOR Game Agent metamodel

tion (defined as a list of declarative Facts). Each plan is composed of actions either directed toward the environment or other agents/actors (causing different type of "effects" on the scenario). In the former case, interactions between Game Agent/human actors (or GA/GA to fully simulate a scenario) are codified by adaptive Dialogues as a set of Sentences (see section 4.2). Figure 2 synthesizes the general structure of the GA model with an UML metamodel.

On a architectural or software level, an agent Engine is defined in order enliven the agents (ie enact the agents life cycle). A Game Agent (GA) begins to update its Facts Base through its perception of the environment (events) and messages from other agents (or players), and will select the appropriate goal according to the situation (role, ...) then select the intention (plan) that will achieve this goal and finally execute the plan. For goal selection, the GA selects first realizable goals by studying the plans feasibi-lity for each goal. Once the realizable goals known, the GA selects the highest priority goal. If there are several goals with the same priority, the GA selects a goal randomly between the highest priority goals. This process enables an agent to adapt its behavior to the state of the environment in a broad sense (ie information about the virtual world and the other actors whether incarnated by agents or not).

The objective of this model is to provide adaptive NPC behaviors, but we also help the scenario designer(s) with an user-friendly and efficient tool to configure these NPCs behaviors. Thus, a graphical editor tool has been developed as illustrated in section 5.1 applied to the risk management case study.

4.1 Action modeling

Actions characterize what an actor can do during a crisis management situation. These actions thus serve as model to design agents' behaviors as well as assessment data when comparing what is done to what should have been done by human actors. Regarding the Game Agent, an action has Preconditions (expressed by a set of facts supposed to be present in the agent Facts Base) and Effects (see fig. 2). An action can be performed in several ways: one shot (action undertaken only once), cyclic (repetitive action such as "check fire progression") or performed at a time t. For each action the designer can specify the number of attempts (if the number is reached without success, the agent goes to the next action).

Actions influence the environment through three kinds of Effects:

Physical effects (PE): such action influences/impacts the SIMFOR 3D environment (as would a

human actor through its user interface).
Knowledge modification effect (KE): direct consequence of an action is facts modification (ie knowledge update) which in turn can abort a goal or validate other goals or actions preconditions.
Message Effect (ME): such effect reflects the social nature of agent and actors as in the SIMFOR SG, they communicate in order to carry out the collaborative task of managing a crisis situation.

Table 1: Example of Crisis Management Actions and their effects.

Action	Effect type	Description
Phone	KE, ME	This action allows joining a player by phone. If both interlocutors are human player, the communication will be oral via VOIP (Voice over IP). If one of the interlocutors is a GA, the communication will be done as a dialogue with textual phrases exchange (see section 5.2).
Fax	KE, ME	This action allows sending a fax to one or more recipients. Faxes are represented by preformatted HTML documents related to crisis management (to be filled in with the right informations).
Radio	KE, ME	This action allows joining a players by radio. The player must then select a channel and press the talk button to communicate with all actors listening to this channel. To add a radio action into a plan, the designer must specify the channel and the id of the dialogue that will be used (by GA).
Talk	KE, ME	This action allows talking with the nearest players (with a defined perimeter).
Move	PE	This action allows to move in the virtual environment. The players can move using the mouse or automatically by selecting a chosen address in address book (either on foot or using a vehicle).
Daybook	KE	the daybook simulates a web portal that allows stakeholders to relay information on the disaster. The player writes the information to be shared and will be available to all stakeholders.

Table 4.2 illustrates actions with various effect as defined in the SIMFOR SG. Some actions may only carried out by GA (NPCs). These actions help to enrich the simulation and make it more realistic to human actors. For example, a (virtual) mayor assistant may prepare a room for a press conference. This action will take some time during which the (human) mayor can not start the press conference. These kinds of actions can also trigger some physical change in the SIMFOR 3D environment, for example activate an emergency siren, and put in place a foam pad (firefighter action). These virtual actions are defined by the scenario designer detailing their execution time, preconditions and effects.

4.2 Dialogue modeling

There are several ways to interact in SIMFOR (phone, fax, radio...) but if one of the interlocutors is a GA, the communication is done in the form of textual dialogue. Partial automated interaction may result in fixed interaction lacking flexibility to reflect the various situations actors and agent may face. In order to avoid that, we have defined a dialogue process as a set of possible Sentences (see fig. 2), each one characterized with a context pertaining to the agents' perception of their environment. A dialogue is thus designed as sentence tree where each node is represented by a sentence. A sentence is characterized by a list of properties described in table 2.

Table 2: Structure of a Sentence.

Sentence Attribute	Description
Id	identify the sentence in the Dialogue tree
parent	list of (possibly) preceding sentences (parent nodes in the Dialogue tree)
Children	list of (possibly) following sentences (children nodes)
Content	represent the displayed text in the dialogue box
Preconditions	contains the preconditions (set of facts) required for the GA to answer this sentence.
Transmitted information	represents the fact transmitted when the phrase is answered (by GA or human player)
Display type	defines if the phrase will be displayed for all roles, only roles in the role list or all roles except the roles in the role list
Roles list	a list of roles involved in the sentence, used for display

In section 5.2, we will see a dialogue example and how the dialogue can be adaptive and interactive.

5. Implementation

In this section we present how was carried out the agent model and the corresponding editor tool, on a software level.

5.1 The agent editor

The agent editor allows the designer to set up the NPCs behaviors. The SIMFOR scenario specifies the actors, means and disasters involved during the game as well as the associated events (fire accident for example). For each SIMFOR actors (ie roles), the designer must associate an NPC. If the actor is not played by a human participant, it will be simulated by a GA, and conversely the GA will be disabled. Moreover, if the game begins with a NPC and a learner want to join in an ongoing game and play the role of the NPC, the GA will be disabled. Also, if a learner plays a role and for some reason leaves the game, the NPC takes over and plays his role.

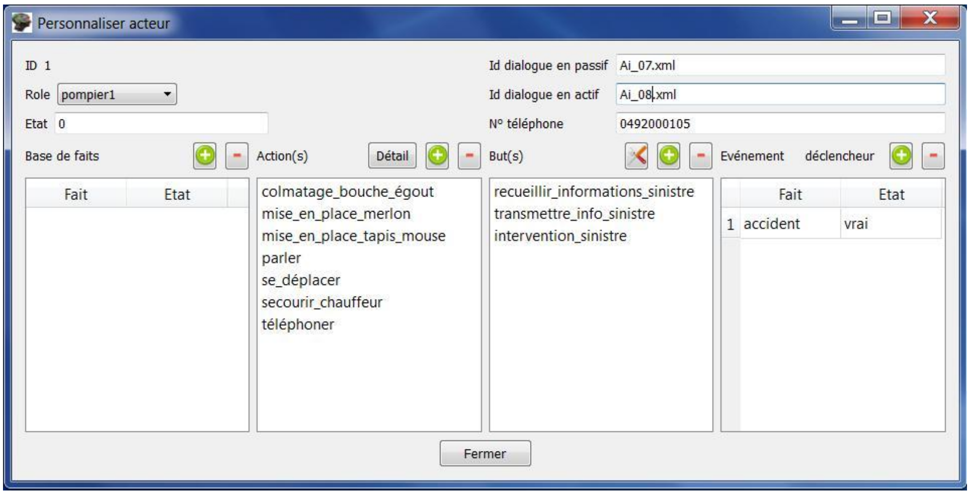

Figure 3: The Game Agent edition.

For each SIMFOR scenario, the designer (domain expert) uses the agent editor to specify facts, actions for each agent (GA) involved in the scenario (figure 3). Facts are used for actions' preconditions and effects, for goal preconditions and for dialogues. The Facts base (Base de faits zone) represents the facts known by the GA at the beginning of the game. Action(s) represent the actions that can be realized by the GA (these actions will be used to define plans). The trigger events (Evenements declencheurs´) are facts which can make Goals realizable (cf. agent engine in section 4). A goal is defined by its name, a priority, preconditions, previous goal (goals that must be previously completed) and finally a set of plans (permitting to achieve this goal). Each plan is represented by a set of (ordered) actions. We will see in more details the dialogues between human players and NPCs in section 5.2.

5.2 The dialogue editor

In crisis management scenario, the different stakeholders must collaborate to restore the normal situation. To do this, there are several interaction processes between actors during the game. In accordance with the Dialogue and Sentence concepts defined in section 4.2, the designer can make rich and interactive dialogue with adaptive response. Within the agent editor, a dialogue editor helps the dialogue and sentence design (figure 4) and saves it in a XML format. We can also import dialogues (of phrase) to create and reuse more complex dialogues.

Figure 5 presents an example of dialogue. This dialogue represents an interaction between the actor Codis and the actor Fireman1. The Codis must inform the fireman1 of the accident, and the fireman1 can answer with two choices: Ask for the road closures or not. The response of Codis actor will depend on its fact base (precondition road blocked). In figure 4 we can see the tree structure of the dialogue as well as the different sentences properties (id, content, precondition etc as described in table 2).

5.3 The agent/environment interaction

The Agent model designed for behaviors simulation is implemented as a library and is completely generic. As agent actions may influence/modify the 3D environment, a communication interface with the SIMFOR 3D environment is required. This communication is based on Commands, which drive the

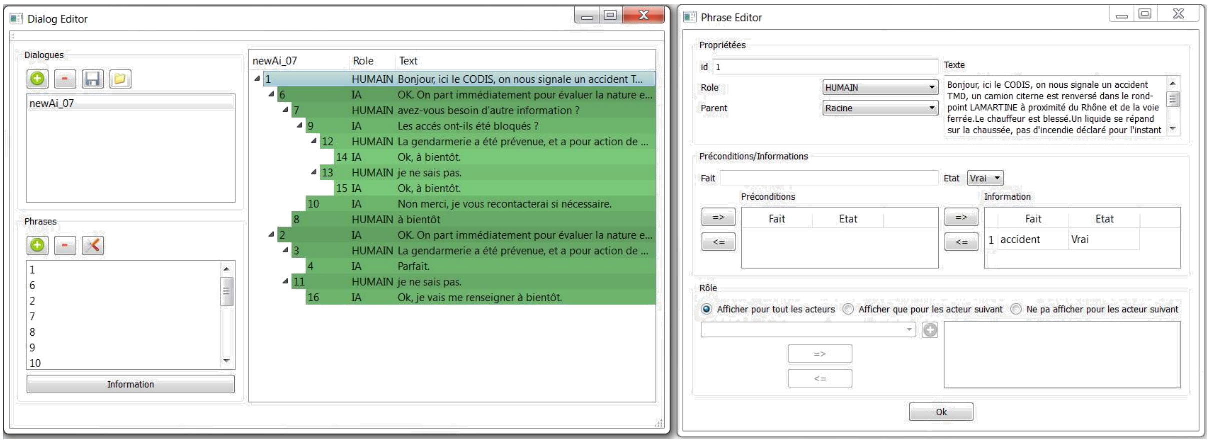

Figure 4: The dialogue and phrase edition.

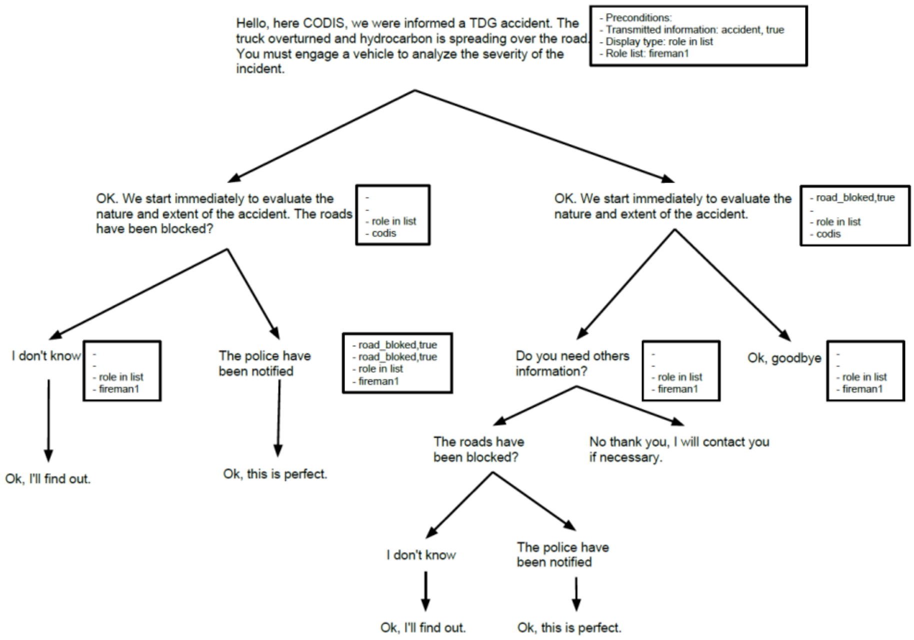

Figure 5: A dialogue example from the TDM scenario.

behaviors simulation model (GA), but are also exploited as learner traces for the learner support and skill assessment.

For example, when the GA wants to perform a move action, the Agent engine sends a Move com-mand with the necessary parameters (destination, means of transport used). SIMFOR processes the request : if the 3D avatar of the GA is near the vehicle and the vehicle is available, SIMFOR carries out the trip (as a 3D animation), otherwise, SIMFOR sends an error command. This process can relate to the MASQ (Multi-Agent Systems based on Quadrants) model (Stratulat et al., 2009). The MASQ model separates the agent mind (decisions) and the agent body (actions). A mind corresponds to the internal structure of an agent or to the decision-making component. The body, either physical or social is parts of the environment and is connected to minds. As with SIMFOR, the mind of the agent is represented by the GA, and the body of the agent is represented by the 3D avatar in the virtual environment.

In the next section, based on the example presented in section 3.1, we illustrate how GA behaviors as well as interactions between players and GA can be edited.

5.4 Case study

To illustrate the behaviors simulation, we present a scenario example defined by a domain expert, which describes the interaction between GA and human player. For this, we resume the scenario presented in section 3.1.

Considering the CODIS role, figure 6 shows an UML activity diagram of the GA. With the given scenario (defined in section 3.1), we have defined and attributed five goals to the GA that play the role of the CODIS.

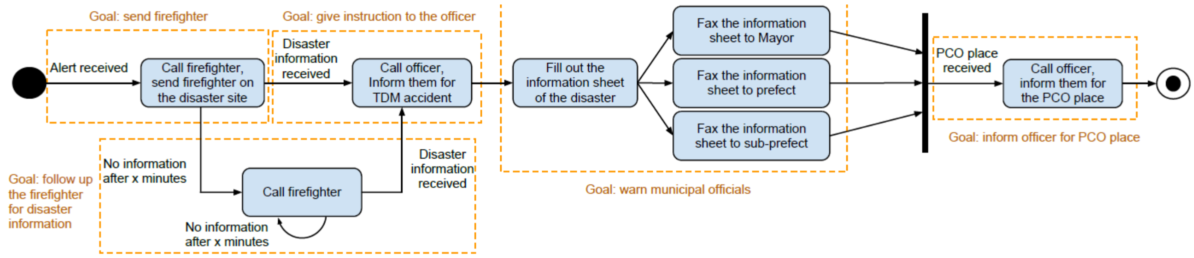

Figure 6: Codis activity diagram

The first goal "send firefighter" is triggered when the CODIS receives the alert (from a witness of the accident), who then sends a firefighter at the disaster scene. This goal is reached by calling the firefighter and ordering him to go to the disaster scene (dialogue process). The second goal "give instruction to the officer" consists in informing the officer of the initial actions to undertake concerning the TDM. The CODIS calls the officer and with a dialogue process gives the procedures to follow. The third goal "Follow up the fire-fighter for disaster information" will be triggered if the firefighter is slow to transmit information. This goal is achieved by a cyclic action repeated until the disaster information is received. The fourth goal "warn municipal officials" consists in warning municipal officials by filling in the information sheet of the disaster and faxing it to the Prefect, Mayor and Sub-prefect. If the CODIS is played by a human player, the information sheet will be filled in manually by the player, if not, the GA will use a pre-filled sheet based on its Facts base content (disaster information). Finally the last goal "inform officer for PCO place", aims at informing the officer of the PCO location, by selecting the right address (found in the Facts base) and calling the officer. This Behavior model can be used for behaviors simulation, but also for learner assessment, as it is part of the domain model (Anderson, 1988) and is used as a reference (overlay model) for the learner model (VanLehn, 1988) (learner's actions and knowledge).

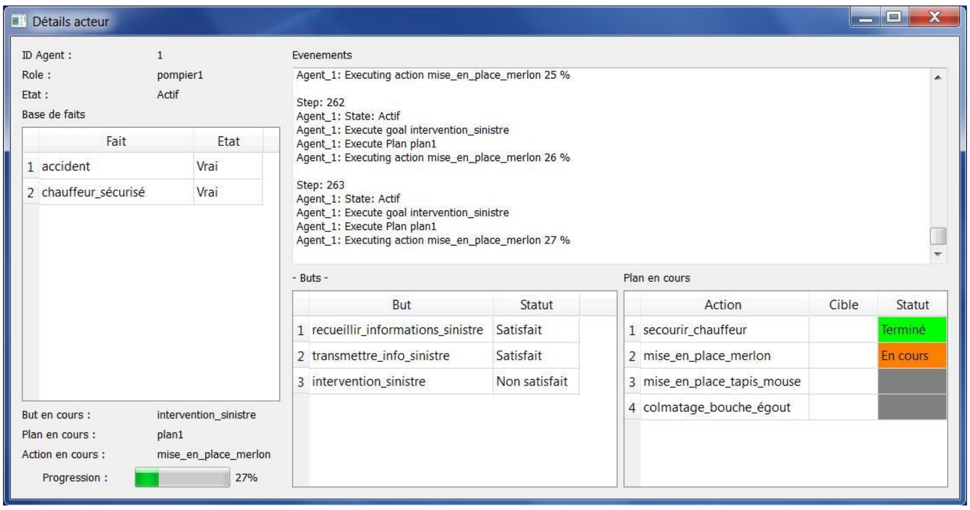

Figure 7: GA monitoring

During the game, we can follow each GA actions (figure 7), its current goal, current plan and the content of its facts base. We can also modify the GA behavior through the interface (figure 7), for example we can reset a goal, add new fact (to trigger some goal), etc. This flexibility in the control of the NPCs is very useful because crisis management is a collaborative process, and the GA behaviors can influence the learner performance. In (Oulhaci et al., 2013b), we have presented different kinds of learner assessment, for example the collective assessment assesses the global performance of all the stakeholders. This performance takes into account the human players (learner) as well as the NPCs.

6. Conclusion

With the growing interest in SG for training purpose, the behaviors simulation of NPCs is increasingly relevant. In this paper, we have presented behaviors simulation could challenge SG into better training simulation. Consequently, in order to address this challenge, we have proposed a BDI-like Game Agent architecture to simulate the NPCs. The goal of this implementation is to cover all trades of crisis management stakeholders and facilitate agent programming for better design of crisis management scenario. This Game Agent model is integrated into SIMFOR project, a serious game for crisis management. The GAs interacts with the SIMFOR environment through command system as well as human players through a dialogues system. This integration is used for behaviors simulation but also plays a role in the learners' assessment. The crisis management is collaborative process, and the learners' player and GA must collaborate to restore the situation. The GAs behaviors can influence learners' performance and the GAs behaviors can be intentionally erroneous to evaluate the learners' behavior in reaction to these errors.

Our immediate work in the SIMFOR project is to focus on the collaborative aspects in the field of crisis management, based on an analysis of the interaction graph permitting real-time interpretation for better pedagogical support. On a more medium-term perspective, primary feedback on our conceptual and architectural proposition reveals sufficient genericity to consider applying our approach to other SG training situation.

Ref´erences´

J. Anderson (1988). 'The expert module'. Foundations of intelligent tutoring systems pp. 21–53.

S. Bakkes, et al. (2009). 'Rapid and reliable adaptation of video game AI'. Computational Intelligence and AI in Games, IEEE Transactions on **1**(2) :93–104.

C. Buche, et al. (2003). 'MASCARET : pedagogical multi-agents systems for virtual environment for training'. In Cyberworlds, 2003. Proceedings. 2003 International Conference on, pp. 423–430. IEEE.

J. F. Hubner,¨ et al. (2002). 'MOISE+ : towards a structural, functional, and deontic model for MAS organization'. In Proceedings of the first international joint conference on Autonomous agents and multiagent systems : part 1, pp. 501–502. ACM.

A. Oulhaci, et al. (2013a). 'Intelligent Tutoring Systems and Serious Game for Crisis Management : a Multi-Agents Integration Architecture'. In IEEE International conference on state-of-the-art research in enabling technologies for collaboration in CT2CM Track. IEEE.

A. Oulhaci, et al. (2013b). 'A Multi-Agent System for Learner Assessment in Serious Games : Application to Learning processes in Crisis Management'. In Seventh IEEE International Conference on Research Challenges in Information Science. IEEE.

A. Rao, et al. (1995). 'BDI agents, From theory to practice'. In Proceedings of the first international conference on multi-agent systems (ICMAS-95), pp. 312–319. San Francisco.

T. Stratulat, et al. (2009). 'MASQ : towards an integral approach to interaction'. In Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2, AAMAS '09, pp. 813–820, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

K. VanLehn (1988). 'Student modeling'. Foundations of intelligent tutoring systems pp. 55–78.

M. Zyda (2005). 'From visual simulation to virtual reality to games'. Computer **38**(9) :25–32.