Introducing autonomous behaviour in instructor-led virtual reality training

M.M. Hendriks[1], S. Hooft[1] and D.J. Garbett[1]
[1]XVR Simulation, Delft, The Netherlands (hendriks@xvrsim.com)

Abstract — XVR Simulation offers advanced incident command training support with immersive 3D virtual reality environments, 2D maps, communication and media-oriented simulators for emergency services, companies, and governments to train their personnel at operational, tactical and strategic command levels. These training tools are explicitly developed to support the flexibility and agility desired by instructors, who are always in full control of the scenario both during its definition as well as during every training session. With customers showing a rising demand to train at a larger scale and include behaviour of large simulated crowds in their scenarios, instructors are being overburdened with orchestrating all the items in response to participants' actions. XVR Simulation has embarked on a quest to include Artificial Intelligence (AI) technologies to assign autonomous adaptive behaviour to any item in any scenario in support of the instructors. The major challenge to be addressed here is to include AI technologies that (a) the instructor can understand, that (b) can handle interventions during training, and (c) are easily used by instructors when creating and running training scenarios.
After investigating possible solutions, XVR Simulation has adopted the MASA DirectAI engine to address the above-mentioned challenge. Within the XVR architecture, the concept of a virtual 'brain' was introduced: a reasoning and decision-making process based on available sensor information that can autonomously decide on the execution of actuators. In order to support understanding and interaction between instructors and 'brains', an appropriate level of abstraction for brains' sensors and actuators needed to be defined. These concepts needed to be integrated into the instructors' mental model of a training scenario. Sensor information is defined in terms of scenario concepts (e.g., opposing party (red team), emergency services (blue team), safe zones, danger zones, victim treatment zones, et cetera). The actuators are defined as scriptable tasks that an instructor creates for a training scenario. Both sensors, actuators and strictly defined decision-making attributes of the brains are freely configurable by the instructor, fulfilling the three criteria of the challenge.
The first implementation of this solution shows that the approach adopted by XVR has resulted in understandable autonomous behaviour for crowds. The chosen solution unburdens the instructors from (a large number of) manual scenario interventions, while still retaining the control necessary in unforeseen training moments.

1 Introduction

XVR Simulation [1] is one of the leading developers of training software for safety and security professionals. Its vision is to create flexible, reliable, content-driven & user-friendly simulation tools where learning is key. These simulation tools, called XVR modules, reside in one XVR platform that allows for diverse and simultaneous learning at schools, training centres, emergency services and private industry & infrastructure. The XVR platform is used in both single- and multi-agency settings for education, training and assessment.

The key driver for the XVR modules is that the instructor is in full control when designing scenarios and during training sessions. While this allows instructors to intervene on participants' actions and, if desired, alter any scenario at run-time, it also poses a burden on them. The instructors are manually adjusting the scenario to provide the appropriate training context for the participants. For run-time scenario adaptations and responsiveness to unforeseen or varying participants' actions there is a need to provide support for the instructor, to alleviate the manual burden of scrolling, selecting, changing properties and assigning tasks and/or actions. Furthermore, the instructors face an increasing workload with the inclusion of large simulated crowds and the trend in larger-scale scenarios and training sessions with a larger number of participants. Key is to reduce the number of distractions for the instructors, so that their focus can remain on the didactical objectives, the participants and their progression through the scenario.

XVR Simulation has set out to investigate the use of Artificial Intelligence (AI) techniques to reduce the instructors' workload. The objective of this investigation is to understand: how to introduce autonomous behaviour that (a) the instructors can understand, that (b) can handle interventions by instructors during a training session, and (c) is easily defined by the instructors when defining scenarios.

2 Current State

Within the XVR platform, the concepts described in table 1 are used.

Table 1. XVR platform concepts

Scenario	an environment with a collection of items and tasks
Environment	a plane of reference in which items are placed: the virtual 3D world
Item	any tangible entity with a visualization that can be placed in an environment: a car, a fire hose, an extinguisher, etc
Avatar	an item which represents a human figure
Crowd	a collection of avatars that appears to be a group
Action	a simple, user-configured, atomic manipulation of an item's generic and/or specific properties, such as move to, hide, embark in vehicle, teleport, or increase fire intensity
Task	a user-defined parameterised sequence of actions that can be invoked on selected items.

The instructor must, during the training, determine appropriate ways to manipulate items, so that the participant perceives reactions on their (in)actions. Often instructors' activities during a training session are:

- Modifying items properties: hide/unhide items, move an avatar, embark an avatar in a vehicle, increase fire intensity, et cetera;
- Pause or completely stop the simulation clock, which pauses all activities (both visual and task related);
- Insert new items in the scenario, in response to a participant's request for additional support;
- Invoke tasks on (pre)selected items, so that these items work in concert. For example, have all firemen embark their vehicle, drive to the designated location, disembark, grab the appropriate equipment and start to extinguish a selected fire. Note that the instructor can decide whether a fire is extinguished or increases in intensity: the instructor is in full control.

An example scenario an instructor is involved in could be a rioting crowd in a small city, where the crowd may behave neutrally or aggressively. The participant, e.g. a riot police commander, is to be trained on handling of the crowd, including the possible dispersion of that crowd.

Figure 1 refers to the aforementioned scenario and zooms in on the interaction with two avatars. One avatar is a rioter, while the other avatar is a riot control officer. The figure focuses on the commands of the participant that causes the riot control officer to approach the rioter. The instructor must both execute the movement of the riot control officer coming closer to the rioter, as well as decide on, and give the instructions for, the rioter to run away from that police officer.

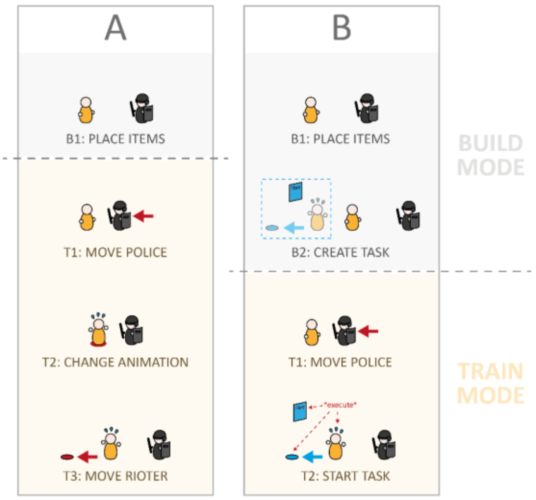

Fig. 1. Instructor manipulation inside current XVR module.

In column A, the instructor manually controls all properties and functionality in real-time. When specifying the scenario, only the 2 avatars need to be created. During the training session, the activities are: the riot control officer moves forward, change rioter avatar animation to fearful, move rioter to destination away from the riot control officer.

In column B, the instructor can use tasks for the same behaviour. Again, both avatars need to be present at specification of the scenario and a task needs to be created with two parameters to let any selected avatar flee to a given destination. During the training session, the riot police officer should move towards the rioter (e.g. via a basic move-to-command given by the instructor). The instructor needs to execute the 'flee' task with parameters of the specific rioter avatar and a specific destination.

Extrapolating these interactions to a rioting crowd results in an additional workload for the instructor. Either the instructor should manually select each individual avatar belonging to a crowd and issue a command (e.g., single avatar changes animation to fearful and runs towards destination X), or define and invoke a task for a group of avatars (e.g., execute 'flee' task for crowd A and destination X'). However, if a crowd needs to move differently than defined, e.g. to run away in multiple directions, this again requires manual assignment of actions and/or tasks.

In sum, the instructor-driven approach to simulation and training has the advantage that the instructor is, indeed, in full control. The instructor decides on the learning experience, and thus has full flexibility to handle expected and unforeseen actions and commands from the participants. The downside is that 'full control' comes with the burden of having to control everything. Although tasks allow for automation of actions, tasks are difficult to adjust during a training session and are difficult to define for all possible participant actions.

3 Design

It is undesirable that the introduction of AI techniques results in the removal of this fine-grained control from the instructor. Rather, these techniques should ideally expand on the current control and assist whenever the instructor handles (too) many property and functionality changes in a given timeframe. This notion is exemplified in figure 2, which outlines the workload for instructors with the autonomous behaviour.

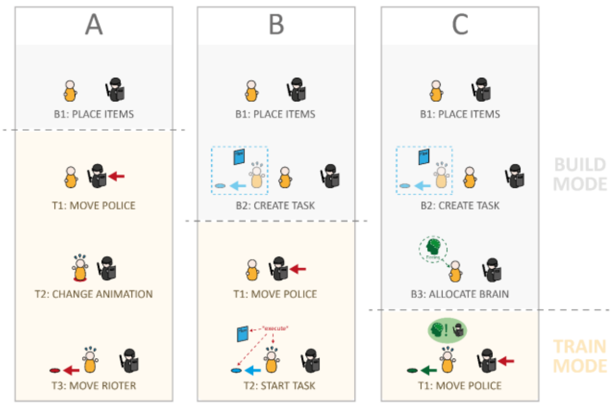

Fig. 2. Addition to the instructor manipulation using autonomous behaviour.

To illustrate this expansion of current control, the instructor interaction, with the example scenario described in the current state, is expanded with a new column introducing the use of brains.

In column C, the expected result of AI support is depicted. During specification of the scenario, the two avatars are placed in the virtual environment, a ‘flee’ task is created and a brain with flee behaviour is assigned to the rioter avatar. During the training session, the riot police officer only needs to be moved towards the rioter. The rest of the behaviour of the rioter is automatically handled by the brain, which uses the ‘flee’ task with the correct parameters.

The approach chosen by XVR Simulation is to integrate the brains in the context of the instructor. In addition, the choice has been to make the instructor’s context leading. The instructor’s context includes concepts related to didactical objectives, scenario-annotations, tasks and other considerations. The approach taken is that sensors of each brain are defined in terms of scenario-concepts. These include not only locations of items, but also annotations such as danger-zone, safety-zone, injury-zone, et cetera. The instructor has the ability to add or remove sensors and change the properties of these sensors in a manner similar to what the instructor is used to, for all item manipulations inside the scenario.

The actuators of each brain are defined in terms of tasks. Although a task may be the same as an (atomic) action (e.g., ‘hide item’ or ‘walk to destination X’), in general tasks are more complicated. A typical ‘flee’ task could, for instance be: change animation to panicking, move with 10 km/hr to a safe destination and once arrived, change to a hide-behind-cover animation. The tasks that serve as actuators can be completely created by the instructor, as long as these adhere to the pre-defined parameters that the brain expects. This allows the instructor to define any specific actuator for any specific scenario (e.g. in scenario A the flee task can be defined to frantically run away, and in scenario B it can be defined to move cautiously). The decision-making capacity of each brain is to decide which task to execute with which set of parameters.

Another concept needed for the shared mental model is behaviour. This concept refers to a certain decision-making capability given sensor information resulting in execution of (complex) actuators. Examples are ‘aggressiveness towards riot police’, or ‘fleeing from danger’, or ‘wandering around’. By exposing ‘behaviour’ as a concept for understanding and interaction among instructor and brains, it becomes possible for an instructor to provide interventions that act on a brain: activate or stop a ‘behaviour’, at runtime include a new ‘behaviour’, et cetera. Similarly, a brain can report on its internal workings by reporting on its active ‘behaviour(s)’ and possible trade-offs in deciding among conflicting ‘behaviours’, thereby exposing some of its internal workings for an instructor as feedback.

The approach described above provides the directions to build an information mapping for the instructor and the brains. As the XVR modules also provide the graphical user interface through which the instructor interacts with the brains, the information mapping is defined for the XVR module and external AI tooling. The expectation is that this information mapping is closest to an instructor’s understanding and expectations of autonomous behaviour.

4 Integration

The XVR platform is built on a sound architectural foundation which imposes additional requirements on the inclusion of any new technology. Regarding the inclusion of autonomous behaviour for items in the virtual world, a clear separation of body and brain is required. With this distinction, any item can be the ‘body’ governed by a brain. The brain then remains a concept outside of the item, maintained by an AI module/tool that facilitates sensor input, reasoning algorithms and actuators to create autonomous behaviour.

DirectAI [1] is a product of MASA Group [2]. DirectAI is a software engine configured by behaviour designers and used to integrate decision processes in simulation agents. A DirectAI brain implements an ‘action selection policy’ for an agent, based on sensory input, drives and representations. This action selection policy is represented in a node hierarchy, consisting of two layers:

1. The ‘decision layer’ propagating decisional information to the action layer;
2. The ‘action layer’ outputting the most adapted action according to the current situation.

This underlying concept of a DirectAI brain closely meets the requirements of the XVR platform and allows for the separation of ‘brain’ (handled by DirectAI) and ‘body’ (handled by the XVR module). The DirectAI brains are supported by the DirectAI runtime environment. This runtime environment needs a coupling with the XVR module via the XVR platform. The integration between these components is depicted in figure 3.

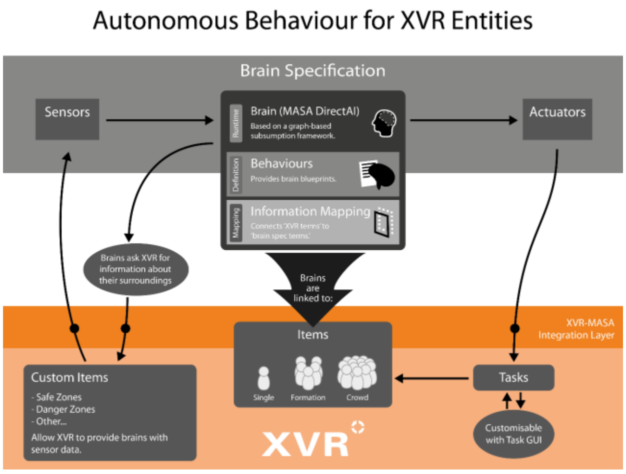

Fig. 3. Integration diagram between XVR platform and DirectAI.

The information mapping is split into two halves: the lowest half (in dark orange) is the integration layer, which exposes XVR module information to and from any DirectAI brain. The upper half (in light grey) is the information mapping where XVR information is interpreted into information and concepts that are of use in the brains’ specifications.

Given this direct dependency between the brains’ specifications and the information mapping with the XVR platform, it is beyond an instructor’s capacity to modify these specifications. It may result in unexpected failures and instability of the XVR module and/or DirectAI runtime environment. Rather, behaviour experts from XVR Simulation and selected partners are specialised in crafting the brains' specifications. These specifications are parameterised and are dependent on specific sensors and actuators. The sensors and actuators (being part of the context of the instructor; e.g. danger zone and a defined 'flee' task) are easily configurable by instructors. During a training session, it is commonplace that tasks fail to execute to completion, e.g. through interventions of other tasks, participants or instructors.

5 Discussion

The chosen AI technology, DirectAI, has been integrated with the XVR platform in accordance with the approach described above. As a result, a number of concepts are defined in addition to the existing concepts of the XVR platform. This shows that the introduction of autonomous behaviour does not alter the product's original concepts and functionality and provides stability for the instructors understanding of the XVR modules.

The integration of the brains with the XVR platform results in bounded flexibility for adapting brains by instructors. An instructor cannot be expected to modify the information mapping or behaviour specification of a brain: this requires highly specialised expertise including extensive testing to assure brains and behaviours of high quality that do not jeopardize the XVR module and DirectAI runtime environment's stability. An instructor can reconfigure brains, by switching behaviours on and off, parameterising behaviours and (re)defining sensors as well as actuators that are used by brains.

The result of including autonomous behaviour in the XVR platform in this way is that the instructor remains in control. Figure 4 shows on the left side the original situation, where the instructor can directly control an item, or use a task to control an item. With the introduction of brains, the instructor can also delegate some control to brains and be in control over the brains as well. The brains are in control of tasks, not of the items directly, allowing for the instructor to regain control easier when needed.

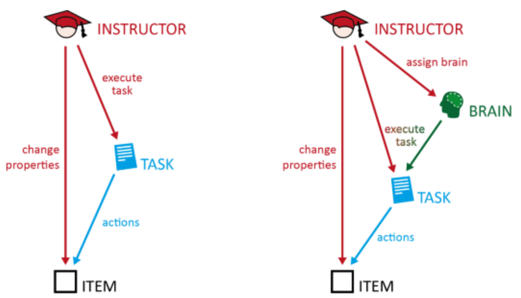

Fig. 4. Instructor remaining in control including autonomous behaviour.

6 Conclusion

This paper describes the solution XVR Simulation chose with the objective of adding autonomous behaviour in support of instructors inside a scenario-based training simulator. The main challenge is summarised as including autonomous behaviour in a fully instructor-defined and run-time customisable training simulator. The DirectAI technology from MASA Group has been integrated into the XVR paradigm and the XVR platform, resulting in:

- an instructor understanding autonomous behaviour of a brain during scenario specification and scenario training. The shared mental model between instructor and brains is based on scenario-concepts already known to and defined by the instructor,
- an instructor's possibility to intervene in the autonomous behaviour of a brain during scenario training,
- an instructor defining and altering the autonomous behaviour of a brain by simple configuration of the behaviours inside the brain, setup of sensors inside the training scenario and definition of the actuators represented as tasks.

Furthermore, the inclusion of the autonomous behaviour is shown to be an extension of the current concepts, thereby easing customer acceptance and ensuring the scalability of the solution inside the XVR platform.

Acknowledgements

The authors are grateful for the discussions, cooperation and general enthusiasm of the many colleagues from XVR Simulation and partners. Special thanks to Yann Prudent, Mathieu Igarza and Fancisco Alberti from MASA Group for the joint work on integrating DirectAI with XVR. Another special thanks goes out to Niek Wijngaards, who was tremendously helpful during both design and documentation phases of this project.

References

[1] XVR Simulation website https://www.xvrsim.com/
[2] MASA Group DirectAI white paper https://masasimulation.files.wordpress.com/2017/12/direct-ai-white-paper-2017.pdf (2017)
[3] MASA Group website https://masasim.com/

Author/Speaker Biographies

Martijn Hendriks is senior developer at XVR Simulation. He holds an MSc. in Media & Knowledge Engineering from Delft University of Technology (NL). His main focus lies in integrating external applications within the XVR platform architecture. **Stephan Hooft** is product manager for XVR Simulation, with an MSc in Design for Interaction, and a BSc in Industrial Design Engineering, both of them earned at Delft University of Technology (NL). He is passionate about understanding and improving the relations between users and products through holistic and grounded designs. **David Garbett** is quality control manager for XVR Simulation, with a BSc in Geography from Coventry University (UK). His responsibilities include management of the QA process and deliveries for XVR simulation.