Agent-Based Simulation as a Tool for the Design of a Virtual Training Environment

Tibor Bosse and Jeroen de Man
Department of Computer Science
VU University Amsterdam
Amsterdam, the Netherlands

Charlotte Gerritsen
Netherlands Institute for the Study of Crime and Law Enforcement
Amsterdam, the Netherlands

***Abstract*—Virtual training is a relatively novel field in which human beings learn to perform certain tasks by repeatedly executing them in a virtual reality environment. To make such training environments more effective, the agent paradigm has proven to be a useful tool. By conceptualising a training system as a 'virtual tutor', the system may be able to support the trainee in a similar manner as human instructors do, among others by providing personalised feedback and adaptive training material. The current paper describes a project that has as aim to develop a virtual training environment for decision making under stress, targeted at professionals in the public domain. The main contribution is two-fold: first, a global overview of the project is presented. Second, a formal approach is put forward for the design of the training environment, based on agent-based simulation and verification. By generating a computational model of the envisioned system and formally analysing the resulting simulation runs, the behaviour of the system can be studied before its actual implementation, thus providing a method for rapid prototyping.**

Keywords—virtual training; simulation; verification.

I. Introduction

In public domains such as law enforcement, public transport, and health care, the professionals involved often have to act under threatening circumstances. For example, in case a tram driver is confronted with an aggressive passenger who refuses to buy a ticket, should she a) confront the aggressor, b) let him go, or c) call for support? Even though they usually have clear instructions about how to act in such situations, these professionals often have difficulties in making appropriate decisions, due to a combination of factors [1, 2]. Firstly, unexpected situations often require improvisation, i.e., deviation from standard protocols. Secondly, time to make decisions is often limited, which makes that the persons experience much pressure. Thirdly, threatening circumstances may cause emotions that are not experienced in everyday situations. Consequently, professionals in the public domain often make suboptimal decisions when they are under stress [2, 3, 4].

In addition, professionals in the public domain have an increased risk of developing anxiety related disorders such as Post-Traumatic Stress Disorder (PTSD), especially if the situation involves extreme violence and/or human casualties [5]. Costs associated with PTSD, both to individuals and to society as a whole, are extremely high [6, 7]. Moreover, even light variants of PTSD may lead to psychological problems, reduced professional ability, or other personal discomfort. For this reason, reducing the number of stress-related disorders in the public domain will save extensive costs and discomfort, both at an individual and a societal level. This leads to the conclusion that there is a strong need for these professionals to be better trained to cope with critical situations. Such training should ideally focus on two aspects, namely 1) improvement of the quality of decision making and 2) better regulation of the emotional response to threatening circumstances.

To deal with threatening circumstances, a variety of techniques and protocols are available that prescribe how employees in domains like law enforcement, public transport, and health care should act under stress [8, 9]. These include communication skills (both at a verbal and a non-verbal level), conflict resolution strategies, and emotion regulation techniques. To learn to effectively apply such techniques, professionals receive dedicated training, both directly on-the-job and in artificial environments. Since on-the-job training is not considered sufficient because of limited possibilities to create the desired learning scenarios, 'offline' training receives much attention. Such training often uses role-play, where the roles are played either by co-students or by professional actors. Although reasonably successful, these types of training have important drawbacks. First, they are very costly, both in terms of money and time. As a result, the frequency by which they are offered is low. And second, there are large differences in the successfulness of role-play-based training: for some students the learning effect is large, whereas for others it is minimal. In conclusion, existing training is expensive, and hard to tailor to individual needs.

Instead, training based on virtual reality (VR) may provide an interesting alternative. The current paper is part of a large project that explores to what extent VR-based training can be used to improve the professional skills of people that have to act in threatening circumstances. VR-based training can be considered a particular instance of serious gaming, i.e., 'the use of computer games designed for a primary purpose other than pure entertainment' [10]. In this case, this purpose would be to acquire better skills with respect to decision making and emotion regulation under stress. Serious games usually involve goal-directed activities in which a competitive element plays a role. Over the last decade, serious gaming has received widespread attention in research and industry, which has resulted in several successful applications. The classical example of a game for training purposes is the flight simulator for aviation pilots, but since then the number of applications has grown rapidly, including serious games for military missions [11], surgery [12] and negotiation [13, 14].

VR-based training offers a solution to many of the problems of real world training. In particular, VR-based training is less costly and time-consuming, easier to set up, manipulate, and repeat, less dependent on place and time, and involves fewer physical risks and ethical difficulties. Moreover, recent developments in Human-Computer Interaction enable training systems to measure the mental and physical state of their users by means of non-intrusive sensors [15]. As a consequence, training can be adapted more easily to the needs of an individual. VR-based training is therefore a promising instrument regarding training of decision making in threatening circumstances, because existing training is costly and difficult to organise.

In spite of this promising prospect, the design of a training system as sketched above is a non-trivial task. It involves the implementation of a number of components, including a 3D virtual environment to generate realistic scenarios, sensors for performing physiological measurements, models for analysis of the trainee's mental state and models for providing personalised feedback. Also, the different components need to be integrated and tested. As a first step in the design of complex software systems, several authors propose (agent-based) simulation as a useful approach (e.g., [16, 17, 18, 19]). By generating a simulation model of the envisioned system and (formally) analysing the resulting simulation runs, the behaviour of the system can be studied before its actual implementation.

Following this line of reasoning, the current paper applies agent-based simulation to study the behaviour of the virtual training environment envisioned in our project. The main contribution is two-fold: first, a global overview of the project itself is presented. Second, a formal approach is put forward for the design of the training environment, based on agent-based simulation and verification. For this second element, the focus is on a specific part of the environment, namely a module that generates feedback on the user's emotional state .

The paper is organised as follows: Section 2 provides an overview of the larger project of which this work is part, and of its current status. In Section 3 we present the modelling approach used in the current paper, based on simulation and formal analysis. In Section 4, a simulation model of the training system is presented, and some resulting simulation runs are discussed in Section 5. In Section 6, the results of the simulation are analysed using formal verification techniques. Section 7 is a conclusion.

II. PROJECT OVERVIEW

The main aim of the project as a whole is to develop an intelligent system that is able to analyse human decision making processes in threatening circumstances, and analyse the causes of incorrect decisions and inadequate stress regulation. The system will be incorporated in an electronic training environment for employees in the public domain, based on Virtual Reality (VR), cf. [20]. In this environment, trainees will be placed in a virtual scenario, in which they have to make difficult decisions, while negative emotions are induced. During the scenario, modern Human Computer Interaction techniques will be applied to measure their emotional state. This information will then be used as input for the intelligent system, to determine why the trainee made certain less optimal decisions and to teach her how to improve this.

An important asset of the VR approach is that the system can adapt various aspects of the training (e.g., scenarios, difficulty level, feedback) at runtime on the basis of its estimation of the trainee's mental state. In this way, both of the training goals can be fulfilled: 1) by selecting training scenarios with an appropriate context in terms of difficulty level, and providing useful feedback, the system can improve the trainee's decision making behaviour, and 2) by selecting training scenarios with an appropriate context in terms of stress level, the system can improve the trainee's emotion regulation skills.

A. *System Overview*

Figure 1 depicts the global architecture of the environment. The rounded rectangles denote components of the system, and the arrows denote information flows. The normal rectangles indicate clusters of components that have the same function (i.e. *analysis* or *support*). In the training environment, the trainee will be engaged in a 3D *virtual reality environment* shown on a computer screen, while being monitored by an intelligent *training agent*. The VR scenario is generated by a separate module within the agent, which contains knowledge about relevant scenarios in a particular domain (e.g., issuing parking tickets, or selling tram tickets). The trainee observes the events that happen in the scenario (via vision and sound), and has to act in the scenario this by selecting the most appropriate action (currently this is simply implemented by means of a multiple choice menu). During training, the human is connected to a (non-intrusive) device[1] that measures physiological states related to arousal and stress; in particular heart rate and skin conductance. The data measured by this device are then used by the agent as input for a computational model that integrates them at runtime, to assess the trainee's levels of stress and (negative) emotions (the *affective model*). This assessment of the trainee's affective state is combined with information about the status of the task (e.g., the actions performed by the trainee), and used by another computational model (the *decision making model*) to assess whether (and why) the trainee made certain mistakes. The outputs of both models are analyses of the trainee's emotional state (e.g., how much stress does she experience?) and of her decision making behaviour (e.g., are any incorrect actions selected?), respectively. This information is used for two purposes: by the *scenario development module*, to modify the running scenario

[1] http://www.biosignalsplux.com/

(e.g., to repeat a certain crucial event, or to induce more stress), and by the *feedback determination module*, to provide feedback to the trainee (e.g., advices to apply a specific emotion regulation strategy); cf. [21].

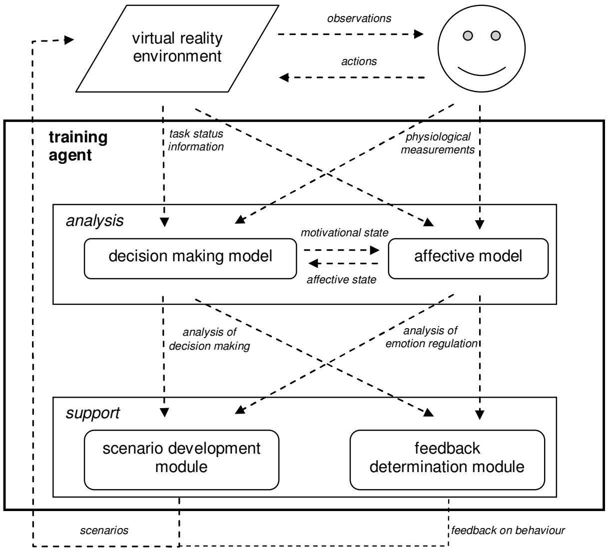

Fig. 1. Global Architecture of the Training Environment

B. *Current Status*

The proposed training environment is being developed and tested in multiple steps, first focussing on separate modules depicted in Figure 1, and finally integrating them into the overall architecture. The current section provides an overview of what has been achieved so far, and of the specific focus of the current paper.

VR Environment and Scenarios

To give trainees the experience that they are dealing with a real-world situation, a realistic *VR Environment* is required. This environment is currently being implemented, in collaboration with a private software company, which develops its proprietary, state-of-the-art game technology. This technology is specifically designed for situation and interaction training. Unlike existing software, it focuses on smaller situations, with high realism and detailed interactions with virtual characters. True-to-life animations and photo-realistic characters help to immerse the trainee in the game. To develop the content for the training, environments, vehicles and other objects are re-created (e.g., training rooms for police officers, or virtual replicas of trams), so that trainees are familiar with their own material and work-environment.

For our project, two different domains are addressed, namely law enforcement and public transport. To create realistic training situations, a library of scenarios has been developed for the targeted domain (e.g., selling tram tickets or handling domestic violence). The content of these training scenarios has been specified based on brainstorming sessions with domain experts. Speech and non-verbal behaviours of the characters are mostly pre-animated, but it is also possible to modify scenarios at runtime, which is done by the *scenario development module*. This results in a game flow that strives for an optimal engagement of the player.

Physiological measurements

VR-based training can only be effective if the virtual scenarios trigger emotional responses that are comparable to the reactions people show to the same stimuli in real world scenarios. To test whether this is indeed the case, a series of pilot experiments has been performed, in which the effect of different types of virtual stimuli on participants' emotional response was investigated. The stimuli used in these experiments varied from affective pictures to affective videos and affective games (all containing material with a negative valence, like guns or scary animals). Although some experiments are still ongoing, preliminary results are promising. For example, the emotional response triggered by the various types of affective stimuli was significantly stronger than the one triggered by 'neutral' stimuli (measured in terms of subjective experience as well as physiological measurements). In addition, 'virtual training sessions' (in which the participants were asked to actively process the material using emotion regulation techniques like reappraisal) seemed to lower the emotional response [22].

Affective model

To enable the training system to draw conclusions about the relation between the VR scenario and the trainee's emotional state, physiological measurements will be processed (at runtime) by an *affective model*. As basis for the affective model, an existing computational model for emotion regulation is used (taken from [23]). Details of this model are explained in Section 4A.

Decision making model

To enable the training system to draw conclusions about the decision made by the trainee, a *decision making model* is being developed. This decision model will incorporate domain-specific knowledge about the task at hand, providing guidelines for how the trainee should act in particular situations. For instance, when dealing with aggressive individuals, de-escalation protocols prescribe a number of consecutive steps that one should apply to calm down the conversation partner [8, 9]. Development of the decision making model is currently in progress; see [24] for a preliminary version. In a later stage, the decision making model will be connected to the affective model, by formalising relations between emotional state and decision making (among others, making use of cognitive *biases* like [25]).

Feedback determination module

As explained above, the purpose of the *feedback determination module* is to provide feedback to the trainee. This feedback may have various forms. First, based on output of the decision making module, the system may provide explanations about when and why incorrect decisions are made. Second, based on output of the affective model, it may provide advice about how to control one's emotional state. For example, the system may advice to pay less attention to a particular negative stimulus (*attentional deployment*) or to re-appraise the meaning of the stimulus (*cognitive change*). The remainder of this paper will focus in particular on the design of the feedback determination module, and more specifically on the second type of feedback - i.e., how to ensure that the trainee reaches a desired emotional state?

III. MODELLING APPROACH

The backbone of the project consists of a generic methodology for development of adaptive intelligent support systems based on computational models of human-related processes. This methodology, of which the details are explained in [15], assumes that all (physiological, cognitive, and social) processes involved in a certain domain can be defined as sequences of *states* over time, and that computational models of such processes can be developed by formalising the temporal relations between those state. This idea corresponds well to the basic assumptions behind the computational modelling and simulation approaches TTL [26] and LEADSTO [27]. Therefore, these two approaches are exploited in the current paper.

The predicate-logical Temporal Trace Language (TTL) [26] integrates qualitative, logical aspects and quantitative, numerical aspects. This integration allows the modeller to exploit both logical and numerical methods for analysis and simulation. Moreover it can be used to express dynamic properties at different levels of aggregation, which makes it well suited both for simulation and logical analysis. To describe dynamic properties of complex processes such as the displacement of crime, explicit reference is made to *time* and to *traces*. A fixed time frame T is assumed which is linearly ordered. Depending on the application, it may be dense (e.g., the real numbers) or discrete (e.g., the set of integers or natural numbers or a finite initial segment of the natural numbers). Dynamic properties can be formulated that relate a state at one point in time to a state at another point in time. A simple example is the following (informally stated) dynamic property about the intensity of an agent's emotional state:

For all traces γ,
there is a time point t such that
agent A experiences an emotion with intensity higher than x.

A *trace* γ over an ontology Ont and time frame T is a mapping $\gamma : T \rightarrow \text{STATES(Ont)}$, i.e., a sequence of states γ_t ($t \in T$) in STATES(Ont). The temporal trace language TTL is built on atoms referring to, e.g., traces, time and state properties. For example, 'in trace γ at time t property p holds' is formalised by state(γ, t) |= p. Typically, state properties refer to agent-based concepts, such as observations, beliefs and desires. Here |= is a predicate symbol in the language, usually used in infix notation, which is comparable to the Holds-predicate in situation calculus. *Dynamic properties* are expressed by temporal statements built using the usual first-order logical connectives (such as ¬, ∧, ∨, ⇒) and quantification (∀ and ∃; for example, over traces, time and state properties). For example, the informally stated dynamic property introduced above is formally expressed as follows:

∀γ:TRACES ∃t:TIME ∃i:REAL
state(γ, t) |= has_emotion(A, i) & i≥x

To be able to perform (pseudo-)experiments, only part of the expressivity of TTL is needed. To this end, the executable LEADSTO language [27] has been defined as a sublanguage of TTL, with the specific purpose to develop simulation models in a declarative manner. In LEADSTO, direct temporal dependencies between two state properties in successive states are modelled by *executable dynamic properties.* The LEADSTO format is defined as follows. Let α and β be state properties as defined above. Then, the notation $\alpha \twoheadrightarrow_{e, f, g, h} \beta$ means:

If state property α holds for a certain time interval with duration g,
then after some delay between e and f
state property β will hold for a certain time interval with duration h.

As an example, the following executable dynamic property states that "if an agent a experiences an emotion with intensity i during 1 time unit, then (after a delay between 0 and 0.5 time units) this agent will experience the same emotion with intensity i*0.9 during 3 time units":

∀a:AGENT ∀i:REAL
has_emotion(a, i) $\twoheadrightarrow_{0, 0.5, 1, 3}$ has_emotion(a, i*0.9)

Based on TTL and LEADSTO, two dedicated pieces of software have been developed. First, the LEADSTO Simulation Environment [27] takes a specification of executable dynamic properties as input, and uses this to generate simulation traces. Second, to automatically analyse the resulting simulation traces, the TTL Checker tool [23] has been developed. This tool takes as input a formula expressed in TTL and a set of traces, and verifies automatically whether the formula holds for the traces. In case the formula does not hold, the checker provides a counter example, i.e., a combination of variable instances for which the check fails.

IV. COMPUTATIONAL MODEL

To enable the training system to reason about the trainee's emotional state, the approach put forward in [15] is used. This approach, which is inspired by the Ambient Intelligence vision [28], assumes that the process under consideration (in this case: human emotion regulation) is modelled as a computational model in LEADSTO (called *domain model*). Next, by applying model-based reasoning techniques to the domain model, an *analysis model* and a *support model* can be created, which can be incorporated into an intelligent system (in this case: the virtual training environment). Based on the analysis model, the system is able to reason about the domain model, in order to draw conclusions about when an undesired state is reached (in this case: when the trainee's emotional state is above a certain threshold). Similarly, based on the support model, the system is able to draw conclusions about which support actions could be effective to deal with the undesired state is reached (in this case: changing the scenario, or providing advice).

In Section 4A, the domain model used for the current project is briefly summarised. The model is an existing computational model for emotion regulation, described in [23]. In Section 4B and 4C, we explain how this domain model is converted to an analysis model and a support model, respectively.

A. Domain Model for Emotional Regulation

The domain model for emotion regulation is taken from [23]. This model, which has been inspired by the theory by Gross [29], is depicted in Figure 2. In the picture the circles represent different states, which are all formalised in a numerical manner, in terms of a variable with a real value between *0* and *1*. In an actual application, real world concepts should be mapped to values in this interval. For instance, a very threatening stimulus (e.g., a person with a gun) could be represented as a *world state* with value 0.9, and a moderately intensive feeling of fear as a *feeling* with value 0.5. Arrows are used to depict the influence of one state on another state. The model that represents the emotion generation is depicted by using solid arrows. The emotion regulation is represented by the *control* state. Each state of the emotion generation model can be regulated by the *control* state (indicated by the dashed arrows). The *control* state has a suppressing effect on the other states. All of the states in the model have a positive effect on the *control* state, representing a kind of monitoring process.

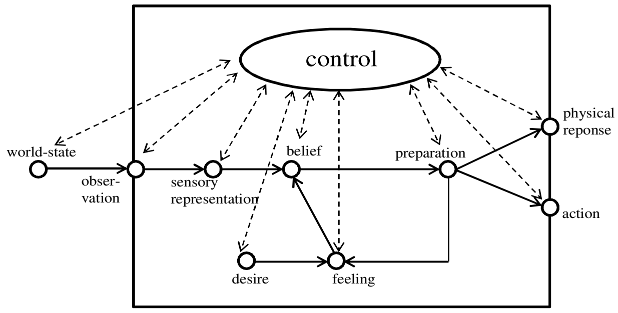

Fig. 2. Overview of the emotion regulation model

As a starting point, the model assumes a *world state* with a negative valence that is observed by the human (e.g., an aggressive individual). This leads to a *sensory representation* and a *belief* about this world state. Next, the person *prepares* to act (e.g., to run away) and this preparation together with a *desire* (e.g., to avoid the stimulus) activate an emotional *feeling*, which in turn may influence the belief state. This generation of feeling from preparation of emotional response follows the account based on an *as-if body loop* introduced by Damasio [30]. In the model, we abstract from the specific 'type' of emotion (e.g., sadness, fear, anger) that is addressed, although we assume that it has a negative valence (we use 'fear' in our examples). Further, the preparation results in both a *physical response* (e.g., increased heart rate) and an *action*.

Each of the states in the emotion generation model is generated by taking the value of that state at time point *t* and adding a fraction of the aggregated mean of values of states that have an influence minus the value of the state at time point *t*. For feeling, the formula would be as follows (where $\eta_{feeling}$ represents the speed of activation spread):

Emotion Generation
∀a,b:REAL
has_value(feeling, b) & aggimpact(feeling, a) $\twoheadrightarrow_{0,\,0,\,\Delta t,\,\Delta t}$
has_value(feeling, b + $\eta_{feeling}$ * [a-b] * Δt)

The aggregated mean in this example would be calculated by using the values of *desire*, *preparation* and (in case of regulation, see below) *feeling.control*:

Aggregated Impact
∀d,p,f,fc,c1,c2,c3,c4:REAL
has_value(desire, d) & has_value(preparation, p) &
has_value(feeling, f) & has_value(feeling.control, fc) &
conn_strength(desire, feeling, c1) &
conn_strength(preparation, feeling, c2) &
conn_strength(feeling, feeling, c3) &
conn_strength(feeling.control, feeling, c4)
$\twoheadrightarrow_{0,\,0,\,\Delta t,\,\Delta t}$
aggimpact(feeling, c1*d + c2*p + c3*f + c4*fc)

The values are weighted by the connection strengths between the two states (see the conn_strength predicates).

The generation of the other states is determined in a similar manner. Only for the *desire* state no aggregated mean has been used since in the model this state is not influenced positively by other states.

To enable the individual to regulate its emotion levels, *control states* have been added for the different states in the model (depicted together as one oval in Figure 2). Each control state has a negative influence on the related state in the emotion generation model. To this end, the value of the downward connections from the control to the different states (i.e., the values of conn_strength(s.control, s)) is taken negative for all state *s*. The upward connections (i.e., conn_strength(s, s.control)) are used to monitor the activation levels of the states. Their strength represents the extent to which the person is able to monitor (and regulate) that particular state. This way, the emotion regulation strategies by Gross [29] can be simulated as follows:

- For *situation selection* and *situation modification* (see [29]), the *world state* is altered (e.g., avoiding a stimulus).
- There are two variants of *attentional deployment* in the proposed model. First, the *observation* state is altered (e.g. by looking away from the stimulus). Secondly, in case of *sensory representation* the internal focus of attention is regulated (e.g., thinking about something else).
- *Cognitive change* is possible when the *belief* is regulated (e.g., reappraisal of the situation: saying to yourself that the situation is not bad). But it is also possible to regulate the *desire* by adjusting one's goals.
- The response-focused regulation strategy *suppression* is applied to *feeling* (e.g., suppressing feelings experienced), *physiological response* (e.g., showing a pokerface) and *preparation/action* (e.g., staying at a location instead of running away).

In [23], a preliminary validation of the model was performed, where the model was used to replicate empirical data obtained from an experiment with human participants [22]. This study pointed out that the model outperformed the performance of a linear approximation.

B. Analysis Model

To convert the domain model sketched above to an analysis model (i.e., a model that allows a system to reason about the dynamics of the domain model), the approach put forward in [15] is used. This approach starts by deciding for which states in the domain model the system can obtain information by 'observing' them. For our domain model, a good candidate is the *world state* (after all, the system already has information about which stimuli are shown, since it generates them itself). Also, the states for *physical response* (using the sensors for heart rate and skin conductance) and *action* (by observing the actions selected in the multiple choice menu) could be 'observed', but in the current paper we do not use them in this way (instead, we reserve them for future validation and tuning purposes, see Section 7). After that, one should decide about which state in the domain model one should draw conclusions. In other words, what is the state of interest of the system? In our case, this is the state *feeling*, since the main goal of the system is to keep the trainee's level of emotion at a certain level. Hence, our system to be designed should receive information about the *world state* as input, and by performing model-based reasoning upon the domain model it should draw conclusions about the *feeling* state. The technique used for this is straightforward: for every formula in the domain model, a corresponding formula for the analysis model is created. The only difference is that in every rule the actual value of a state is replaced by a belief of the system about this value:

Beliefs on Emotion Generation
∀a,b:REAL
belief(feeling, b) & aggimpact(feeling, a) $\rightarrow_{0,\,0,\,\Delta t,\,\Delta t}$
belief(feeling, b + $\eta_{feeling}$ * [a-b] * Δt)

Also, a constant f_{max} is introduced, which specifies the maximal value of the *feeling* state that is still considered desirable by the system (i.e., a desire of the system about the human's emotion). Finally, a formula is introduced to assess whenever an undesired situation is detected, i.e. when the believed value of the *feeling* state exceeds the desired value:

Assessments
∀b:REAL
belief(feeling, b) & desire(feeling, f_{max}) & b>f_{max} $\rightarrow_{0,\,0,\,\Delta t,\,\Delta t}$
assessment(feeling)

C. Support Model

The analysis model enables the system to assess when the human's state of feeling has reached an undesirable level. Once this is the case, the support model is activated. As discussed earlier, in theory various forms of support can be provided, and one of the tasks of the support model is to reason about which type of support is most appropriate. To this end, it can again reason through the domain model, to calculate the expected effect of several hypothetical support actions, and select the most appropriate one. The types of support that can be provided include the following (see the corresponding strategies in Section 4A):

- *Situation modification*: modifying the stimuli displayed in the game (e.g., make a character show less aggressive behaviour).
- *Attentional deployment*: advising the trainee to focus her attention on something else (e.g., count to 10).
- *Cognitive change*: advising the trainee to re-interpret the negative stimulus (e.g., trying to show more understanding for the aggressive person's situation).
- *Suppression*: advising the trainee to suppress her negative feelings or tendencies.

Some examples of simple formulas that model situation modification and cognitive change are the following (where *w1* is a parameter that can either be given a static value or be determined at runtime based on the previous value):

Situation Modification
assessment(feeling) $\rightarrow_{0,\,0,\,\Delta t,\,\Delta t}$
set_value_to(world_state, w1)

Cognitive Change
assessment(feeling) $\rightarrow_{0,\,0,\,\Delta t,\,\Delta t}$
propose(cognitive_change)

Suppression
assessment(feeling) $\rightarrow_{0,\,0,\,\Delta t,\,\Delta t}$
propose(suppression)

Furthermore, the effect of a particular strategy on the human should be simulated. For example, for cognitive change, this could be realised by adapting the connection strength from control to belief (where *w2* is a parameter):

Effect of Cognitive Change
propose(cognitive_change) $\rightarrow_{0,\,0,\,\Delta t,\,\Delta t}$
set_value_to(conn_strength(belief.control, belief), w2)

The following sections illustrate the impact of these support actions on the human's level of emotion via some simulations.

V. SIMULATION RESULTS

To study the behaviour of the model described in the previous section, a number of simulation traces have been generated under different parameter settings. The current section describes four illustrative examples of such simulation traces in more detail. The results of the simulation traces are depicted in Figure 3. In this figure, time is shown on the horizontal axis and the activation values of the state *feeling* (under different circumstances) are shown on the vertical axis.

In all simulations, the activation values of the *world state* and the *desire* have been set to 0.8. This indicates, respectively, that the trainee is exposed to a rather strong stimulus and has a strong desire to avoid this stimulus. The other activation states all start with a value of 0. All downward *connection strengths* from the control component to the regular states have been set to -0.1. All η values (for speed of activation spread) have been set to 0.1. The training agent makes use of the same parameter settings.

In all simulations, except for the *no support* model, the training agent can provide support, as explained in the previous section. To this end, the feeling threshold f_{max} has been set to 0.5. If this threshold is exceeded, the supervisor agent will provide support. Besides the no support case, three different situations are shown, namely support via *situation modification*, *cognitive change*, and *suppression*. Here, the support parameters *w1* and *w2* have been set to 0.2 and -1.

In Figure 3 the blue line shows the situation in which there is *no support*. In this case, the experienced feeling increases until it reaches an equilibrium of 0.6 around time point 80. The green line represents the situation in which the *cognitive change* strategy has been applied. In this situation the activation value increases for a while, but about halfway the simulation this value exceeds the threshold f_{max}, which makes that the agent starts providing support (i.e., telling the human to perform reappraisal). As a result, the emotion intensity drops quickly below this value, after which it stabilises just below 0.4 around time point 100. Similarly, the agent can support the human by decreasing the emotional valence of the stimuli (*situation modification*, red line). In that case, the value of the feeling increases until f_{max} is exceeded. Then, because the intensity of the emotional stimulus is reduced permanently, the value keeps decreasing gradually. Finally, a situation in which the agent tells the human to *suppress* its response shows similar results to the situation in which no support was provided. The reason for this is that, although the human does suppress her physiological response, this does not have any effect on the experienced feeling.

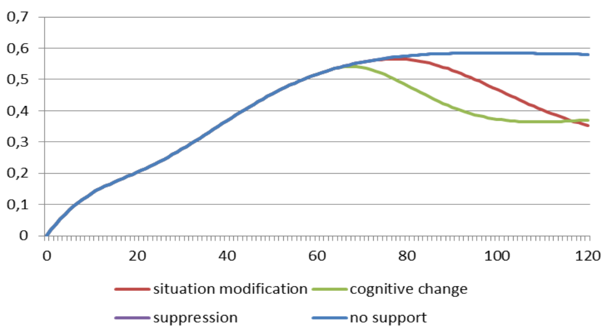

Fig. 3. Example Simulation Traces

VI. Formal Analysis

Using dynamic properties specified in the Temporal Trace Language as explained in Section 3, a formal analysis of the model has been made. A total of 6 dynamic properties are used to verify the ability of our support model to lower a heightened *feeling* by proposing a particular type of support. These properties are specified in a hierarchical fashion, such that when the global property fails, the cause of this failure can be found by systematically checking lower level properties until the fault has been identified. In particular, the following implications between dynamic properties hold:

IP1 & IP2 ⇒ GP1
LP1 & LP2 & LP3 ⇒ IP2

At a global level, the agent needs to guarantee that, whenever the level of *feeling* gets too high (i.e. above a certain threshold), then later this *feeling* returns to a level below this threshold. This has been specified in global property GP1.

GP1 Feeling always returns below threshold
∀γ:TRACES ∀t1:TIME ∀x1:REAL
state(γ, t1) |= feeling(x1) & x1>feeling_threshold ⇒
[∃t2:TIME ∃x2:REAL
state(γ, t2) |= t2>t1 & feeling(x2) & x2<feeling_threshold]

When this property fails, it is important to know whether this is caused by a failure to provide support or if the support provided did not succeed in lowering the *feeling*. These intermediate properties have been specified below in IP1 and IP2. As can be seen, it follows logically that when GP1 fails, either IP1 or IP2 also needs to fail, thereby identifying the cause more precisely.

IP1 Feeling above threshold triggers support
∀γ:TRACES ∀t1:TIME ∀x1:REAL
state(γ, t1) |= feeling(x1) & x1>feeling_threshold ⇒
[∃t2:TIME ∃s:SUPPORT
state(γ, t2) |= t2≥t1 & propose(s)]

IP2 Support lowers the feeling
∀γ:TRACES ∀t1:TIME ∀ s:SUPPORT
state(γ, t1) |= propose(s) ⇒
[∃t2:TIME ∃x:REAL
state(γ, t2) |= t2>t1 & feeling(x) & x<feeling_threshold]

Considering the simulation in Figure 3 where no support is given, GP1 fails and subsequently IP1 also fails, thereby identifying the problem in that simulation. However, if a situation is considered whereby for example *suppression* is proposed, which is expected not to lower the *feeling* but only the *physical response*, IP2 would fail. In that case, a last set of local properties (LP1-LP3) has been defined to check whether an undesired type of support has been proposed or that the support failed in lowering the *feeling*.

LP1 Support implies situation selection or cognitive change
∀γ:TRACES ∀t1:TIME ∀s:SUPPORT
state(γ, t1) |= propose(s) ⇒
[∃γ:TRACES ∃t0,t2:TIME ∃x0,x2:REAL
(state(γ, t0) |= world_state(x0) & t0<t1 &
state(γ, t2) |= world_state(x2) & t2≥t1 & x2<x0) ∨
(state(γ, t0) |= conn_strength(belief.control, belief), x0) & t0<t1 &
state(γ, t2) |= conn_strength(belief.control, belief), x2) & t2≥t1 &
x2<x0)]

LP2a Situation selection lowers the feeling
∀γ:TRACES ∀t1,t2:TIME ∀x1,x2:REAL
state(γ, t1) |= world_state(x1) &
state(γ, t2) |= world_state(x2) & t2>t1 & x2<x1 ⇒
[∃t3:TIME ∃x:REAL
state(γ, t3) |= t3≥t2 & feeling(x) & x<feeling_threshold]

LP2b Cognitive change lowers the feeling
∀γ:TRACES ∀t1,t2:TIME ∀x1,x2:REAL
state(γ, t1) |= conn_strength(belief.control, belief), x1) &
state(γ, t2) |= conn_strength(belief.control, belief), x2) &
t2>t1 & x2<x1 ⇒
[∃t3:TIME ∃x:REAL
state(γ, t3) |= t3≥t2 & feeling(x) & x<feeling_threshold]

Proposing *suppression* would make LP1 fail, showing that the support provided does not have effects that are expected to lower the *feeling*. Table 1 gives on overview of which properties were satisfied (✓), not satisfied (✗), or not checked (-) against the different scenario's. At this moment, the analysis has been made on a support model operating on a simulation of the domain model, making it impossible for LP2a or LP2b to fail. However, by including these properties, similar analyses can be made when such an agent operates on a real human to identify whether certain types of support do not work for particular people.

TABLE I. ANALYSIS OF DIFFERENT SCENARIO'S USING TTL PROPERTIES

Scenario	GP1	IP1	IP2	LP1	LP2a	LP2b
No support	✗	✗	✓	-	-	-
Situation selection	✓	-	-	-	-	-
Cognitive change	✓	-	-	-	-	-
Suppression	✗	✓	✗	✗	✓	✓

VII. CONCLUSION

The paper presented a research project in which a virtual training environment for decision making under stress is being. Furthermore, a formal method was put forward for designing and analysing such a system. For this purpose, a domain model on emotion regulation has been described and used for developing analysis and support models. Simulations of these models were performed to illustrate the usage of dynamic properties to analyse and verify the system's behaviour.

Using this method, it is possible to formally analyse the performance of the analysis and support models, without the need to involve human participants. This makes it possible to develop and test multiple prototypes (i.e., simulated instances of the virtual training environment) rapidly. Obviously, in a later stadium, it is important to include human participants in the development cycle. But even then, the same dynamic properties can be used to verify the system performance. At that point, there may be some error between the values predicted by the agent and those measured in the real world. Therefore, a parameter adaptation model is currently being developed, providing the training agent with a means to tune relevant parameter values by machine learning.

There are of course more steps that need to be made before the training environment can be completed. Besides finalising the decision making model, all components need to be combined and implemented in the final virtual environment. After this, experiments with professionals using the virtual training will provide the data to evaluate the overall system.

ACKNOWLEDGEMENTS

This research was supported by funding from the National Initiative Brain and Cognition, coordinated by the Netherlands Organisation for Scientific Research (NWO), under grant agreement No. 056-25-013

REFERENCES

[1] M.R. Endsley (1997). The role of situation awareness in naturalistic decision making. Naturalistic decision making: 269-284. Mahwah, NJ: Erlbaum,.

[2] G.F. Loewenstein, and J. S. Lerner (2002). The role of affect in decision making. In: R. Davidson, K. Scherer, & H. Goldsmith (Eds.), Handbook of affective science, pp. 619–642. New York: Oxford University Press.

[3] G.L. Clore, and J. R. Huntsinger (2007). How emotions inform judgement and regulate thought. Trends in Cognitive Sciences, vol. 11, pp. 393-399.

[4] F. Ozel (2001). Time pressure and stress as a factor during emergency egress. Safety Science, vol. 38, pp. 95-107.

[5] C. R. Brewin, B. Andrews, and J. D. Valentine (2000). Meta-analysis of risk factors for posttraumatic stress disorder in trauma-exposed adults. Journal of Consulting and Clinical Psychology, vol. 68, issue 5, pp. 748-766.

[6] R. C. Kessler (2000). Posttraumatic Stress Disorder: The Burden to the Individual and to Society. Journal of Clinical Psychiatry, vol. 61 (suppl. 5), pp. 4-12.

[7] H.U. Wittchen and F. Jacobi (2005). Size and burden of mental disorders in Europe – a critical review and appraisal of 27 studies. European Neuropsychopharmacology, vol. 15, issue 4, pp. 357-376.

[8] L.N. Anderson, and J.T. Clarke (1996). De-escalating verbal aggression in primary care settings. Nurse Pract. 21(10):95, 98, 101-2.

[9] G.M. Breakwell (1997). Coping with Aggressive Behaviour. J. Wiley and Sons.

[10] U. Ritterfeld, M. Cody, and P. Vorderer (eds.) (2009). Serious Games: Mechanisms and Effects. New York/London: Routledge.

[11] A. Van der Hulst, T. Muller, S. Besselink, D. Coetsier, and C. Roos (2008). Bloody serious gaming: experiences with job oriented training. In Proceedings of I/ITSEC '08, pp 375 - 385, Orlando, Fla, USA.

[12] M. Graafland, J.M. Schraagen, and M.P. Schijven (2012). Systematic review of serious games for medical education and surgical skills training. The British journal of surgery 99(10):1322-30.

[13] J. Kim, R.W. Hill, P. Durlach, H.C. Lane, E. Forbell, C. Core, S. Marsella, D. Pynadeth, and J. Hart (2009) BiLAT: A game-based environment for practicing negotiation in a cultural context. International Journal of Artificial Intelligence in Education, vol. 19, issue 3, pp. 289-308.

[14] J. Broekens, M. Harbers, W. Brinkman, C.M. Jonker, K. van den Bosch and J.CH. Meyer (2012). Virtual Reality Negotiation Training Increases Negotiation Knowledge and Skill. In: Proceedings of IVA'12, pp. 218-230.

[15] T. Bosse, M. Hoogendoorn, M.C.A. Klein, M.C.A., R. van Lambalgen, P.-P. van Maanen, and J. Treur (2011). Incorporating Human Aspects in Ambient Intelligence and Smart Environments. In: Chong, N.Y. and Mastrogiovanni, F. (eds.), Handbook of Research on Ambient Intelligence and Smart Environments: Trends and Perspectives. IGI Global, pp. 128-164.

[16] A.M. Christie (1999). Simulation: An Enabling Technology in Software Engineering. CROSSTALK - J. of Defense Software Engineering, pp. 2-7.

[17] M.I. Keller, R.J. Madachy, and D.M. Raffo (1999). Software Process Simulation Modeling: Why? What? How? J. of Systems and Software, Vol.46, No.2/3.

[18] F. Lin, C.J. Su and M.M. Tseng (1999). An Agent-based Approach to Developing Intelligent Virtual Reality-based Training Systems. In: Proc. of the 11th Int. Conf. on Tools with Artificial Intelligence, IEEE Computer Society, pp. 253-260.

[19] D. Richard and J. Porte (2009). Developing an Agent-based Training Simulation using Game and Virtual Reality Software: experience report. In: Proceedings of the 6th Australasian Conference on Interactive Entertainment. ACM Press: New York.

[20] S. Marsella, and J. Gratch (2002). Modeling the influence of emotion on belief for virtual training simulations. In: Proceedings of the 11th Conference on Computer-Generated Forces and Behavior Representation, Orlando, FL, May 2002.

[21] A. Heuvelink and T. Mioch (2008). FeGA: A cognitive Feedback Generating Agent. In Proc. of the 7th IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT 2008) (pp. 567-572). IEEE Computer Society Press.

[22] T. Bosse, C. Gerritsen, J. de Man, and J. Treur (2012). Measuring Stress-Reducing Effects of Virtual Training Based on Subjective Response. In: Proceedings of ICONIP'12. Lecture Notes in Artificial Intelligence, Springer Verlag, pp. 322-330.

[23] T. Bosse, C. Gerritsen, J. de Man, and J. Treur, J. (2013). Learning Emotion Regulation Strategies: A Cognitive Agent Model. In: Proc. of the 13th Int. Conf. on Intelligent Agent Technology, IAT'13. IEEE CS Press, pp. 245-252.

[24] T. Bosse and S. Provoost (2014). Towards Aggression De-escalation Training with Virtual Agents: A Computational Model. In: Proceedings of the 16th International Conference on Human-Computer Interaction, HCI'14. Springer Verlag, in press.

[25] A.G. Brand (1987). Hot Cognition: emotions and writing behaviour. Journal of advanced composition, vol. 6 issue 28.

[26] T. Bosse, C.M. Jonker, L. Van der Meij, A. Sharpanskykh and J. Treur (2009). Specification and Verification of Dynamics in Agent Models. International Journal on Cooperative Information Systems, vol. 18, pp. 167-193.

[27] T. Bosse, C.M. Jonker, L. Van der Meij, and J. Treur (2007). A Language and Environment for Analysis of Dynamics by Simulation. Journal of Artificial Intelligence Tools, vol. 16, issue 3, pp. 435-464.

[28] E. Aarts, R. Harwig and M. Schuurmans (2001). Ambient Intelligence. In: P. Denning (ed.), The Invisible Future, McGraw Hill, New York, pp. 235-250.

[29] J.J. Gross (2001). Emotion regulation in adulthood: timing is everything. Current directions in psychological science, 10(6):214-219, December 2001.

[30] A.R. Damasio (1999). Descartes' Error: Emotion, Reason and the Human Brain. Papermac, London, 1994.