Reinforcement Learning as a tool to make people move to a specific location in Immersive Virtual Reality

Aitor Rovira[a,b], Mel Slater[c,b]

[a] *Nara Institute of Science and Technology, Japan*
[b] *University College London, United Kingdom*
[c] *ICREA-University of Barcelona, Spain*

Abstract

This paper describes the use of Reinforcement Learning in Immersive Virtual Reality to make a person move to a specific location in a virtual environment. Reinforcement Learning is a sub-area in Machine Learning in which an active entity called *agent* interacts with its environment and learns how to act in order to achieve a pre-determined goal. The Reinforcement Learning had no prior model of behaviour and the participants no prior knowledge that their task was to move to and stay in a specific place. The participants were placed in a virtual environment where they had to avoid collisions with virtual projectiles. Following each projectile the agent analysed the movement made by the participant to determine paths of future projectiles in order to increase the chance of driving participants to the goal position and make them stay there as long as possible. The experiment was carried out with 30 participants, 10 were guided towards the leftmost part of the environment, 10 to the rightmost area, and 10 were used as control group where the projectiles were shot randomly throughout the game. Our results show that people tended

Email address: aitor@is.naist.jp (Aitor Rovira)

Preprint submitted to International Journal of Human-Computer Studies April 24, 2016

to stay close to the target area in both the Left and Right conditions, but not in the Random condition.
Keywords: Immersive Virtual Reality, Reinforcement Learning

---

1. Introduction

Normally Reinforcement Learning (RL) is used in computer graphics and virtual reality to control the behaviour of characters, for example, so that they walk, run, jump, avoid obstacles, and appear to do this with the most humanlike behaviour possible (Lee and Lee, 2006; Treuille et al., 2007). The aim in this study, however, was to use RL to influence the behaviour of people in an Immersive Virtual Environment (IVE) where the RL agent would learn to guide them to carry out a task of which they were unaware. This technique relies on the participants exhibiting Presence, that is, responding realistically to the virtual situation and events (Sanchez-Vives and Slater, 2005). In earlier work, Kastanis and Slater (2012) showed a novel way to use RL to elicit a required behaviour from people by taking advantage of Proxemics (Hall, 1966). In that study a virtual character could move closer to, away from or wave to the participant to come closer. The goal of the RL was to get the person to go backwards compared to their starting point to a specific position. It took no more than 7 minutes for the RL to learn to make people move to the target location. However, this was a one-dimensional problem. In our study, we allowed people free movement in a two dimensional area using natural movements of the body, such as walking or running.

Our design executed in an IVE (a Cave), consisted of a game-type scenario where the participant needed to move around in order to avoid being hit by

virtual projectiles shot from a virtual spacecraft, controlled by the computer. At the same time, and without the participant's knowledge, the RL agent analysed the movements of the person following each projectile. Its goal was to make the person move to a target location and make them stay there as long as possible. Our hypothesis was that, with no prior knowledge for each participant and given enough time to try a reasonable number of actions, the RL agent would learn to make people move to a specific location in the virtual environment and stay there the longest time possible. On the other hand, an agent shooting randomly during all the game would not achieve the same results.

The contribution of this study is to show with a simple experiment, the potential applications of using RL to influence in people's behaviour individually and change their responses without prior knowledge about how a person behaves. This study uses a virtual environment as an example, but the applications can be easily extended to other human-computer interaction fields such as websites, for example studying the users' behaviour and their interests to increase the number of web pages visited.

The remainder of this section is dedicated to a description of RL and IVR, and gives some examples of relevant studies in these two areas. In the Methods section, we describe the design of the study and its procedures. The Results section contains the statistical analysis carried out with the data collected. The Discussion section summarises the findings discovered previously, expands on the research topic, and suggests ideas to be considered in the future.

1.1. Reinforcement Learning

Reinforcement Learning (RL) is a method of Machine Learning that tries to solve problems designed as a Markov Decision Process. The typical setup involves an active entity called *agent* that interacts with its environment. Given a current state, the agent takes an action and observes the changes in the environment. During this process, the agent might get a positive or negative reward in the form of a numerical value. In RL, the agent needs to develop a strategy to maximise its long term reward (Kaelbling et al., 1996; Sutton and Barto, 1998; Wiering and van Otterlo, 2012), usually within a limited amount of time. As the agent tries different actions, it builds up a statistical model that determines the best action to take for each possible state individually that will help the agent to achieve it.

RL has some substantial differences with Supervised Learning. Supervised Learning algorithms have two sequential stages, learning and exploiting the knowledge. RL problem do not necessarily have a learning stage in the beginning, the agent carries out both tasks concurrently. The experience collected in previous interactions with the environment can be used as knowledge as soon as it is obtained. Supervised Learning also relies on an external entity that knows a priori the right solution and teaches the agent in the learning stage. In a RL problem, this information is often not available and the agent is able to learn without any prior experience and without knowing anything about the goal. Moreover, if there is a change in the environment, a RL agent may be able to adapt its strategy. The agent might be required to carry out the same task periodically. Every time it performs the task, it uses the experience obtained in previous episodes. But it might need to perform

the task only once. In this case, the strategy needs to be updated on the fly, as soon as results are obtained.

The design of the RL problem is the critical step for a RL setup to be successful. While we could say that a RL agent will find the optimal solution if there is no time limit, it is sometimes necessary that the optimal solution is found within a reasonable number of trials. A greater number of actions and state variables can increase the accuracy of the model of the environment, but it also entails an exponential growth of state-action combinations to try. On the other hand, using a simplified version can make the design inaccurate. The idea is to find a good balance between available time and complexity of combinations.

In a deterministic environment, the reward for each action-state pair remains constant. The reward obtained can also depend on a probability function, which allows the agent to try various combinations at different times to adjust the policy. Furthermore, the environment can be dynamic and change over the time in a way that can not be predicted. In this case, the agent has to find a good trade-off between exploiting the knowledge expecting to obtain a high value immediate reward or explore and observe if there was any change.

The first successful applications of RL were used to train a machine to learn to play board games. Board games provide a discrete and finite deterministic environment ideal for simple RL problems. They are also repeatable, which means that the agent can play as many games as needed and accumulate the knowledge obtained based on the outcome of each game. After a computer successfully learned to play checkers (Samuel, 1959, 1967), other

board games followed afterwards, such as Chess, Go and Othello.

More recently, it has been applied to computer games in more complex setups, for example, affording a computer to learn how to improve its skills playing a role-playing game (Spronck et al., 2003) or commanding an entire army with the use of various agents concurrently (Marthi et al., 2005) or even an agent learning to play different games (Mnih et al., 2015). Robotics is the other major field where RL has been applied, where it has been used to make mechanical devices learn to perform physical tasks (Kober et al., 2012; Kormushev et al., 2013). While the problem of dimensionality is also present in these fields due to large number of degrees of freedom, Robotics has the added difficulty of the accuracy of the sensors and actuators employed. Other applications of RL are in systems control. One example is computer animation, where a RL agent can learn to find the path to a target position (Vigorito, 2007), in environments with obstacles (Treuille et al., 2007; Kolter and Ng, 2009) which can be useful for autonomous entities such as virtual characters or unmanned aerial vehicles (Ng et al., 2004; Hoffmann et al., 2005).

1.2. Immersive Virtual Reality

IVR allows the realization of scenarios in a laboratory environment where the responses can be observed and recorded in a controlled situation. It also supports repeatability for as many participants as needed for each study. Moreover, people tend to have authentic responses in IVR if certain technical requirements are met. These requirements include a low latency tracking system (Meehan et al., 2003) to adjust the imagery to the person's perspective, and a stereoscopic display (IJsselsteijn et al., 2001) with a minimum

required field of view degree (Lin et al., 2002). These technological requirements allow the participant to perceive and, to some extent, interact with the environment in a realistic way that the results of their actions are contingent with their expectations (Noe, 2004). When this happens, people tend to have the feeling of being in the place depicted, even knowing that they are experiencing a computer generated simulation. This is referred to in the literature as the sense of Presence (Held and Durlach, 1991; Slater et al., 1994; Sanchez-Vives and Slater, 2005).

A wide range of applications have been developed over the last two decades to study people's behaviour in situations that can be easily controlled and manipulated in an IVR system. Some examples of these are the study of violence emergencies Slater et al. (2013), therapy related studies such as treatment of phobias (Pertaub et al., 2002; Garcia-Palacios et al., 2002), and Post-Traumatic Stress Disorder (Rothbaum et al., 1999). But all these scenarios are prescribed or have little interactivity, and have been implemented to observe people's responses to a scripted situation. Besides this, people might have different reactions based on their personality traits. Therefore, a certain degree of adaptability can be useful in these situations. Kastanis and Slater (2012) used RL to learn how to make every individual achieve a goal in the virtual environment, without the use of any previous knowledge by the RL observing how participants responded to the actions of a virtual character that it controlled. The participant was placed in an alley and the goal was to make them move to a location that was behind them by only using the principle of Proxemics (Hall, 1966) so that participants would tend to move backwards away from the virtual character when

it invaded their personal space. This study showed how to change the events depending on the real person's behaviour and regardless of how other participants performed. However, the participant's movements were limited to one dimension and the RL agent could only choose from a set of 4 actions, move forward, move backwards, stay idle or call the participant to move towards it.

2. Methods

2.1. Scenario

In the scenario, the spacecraft could move left and right for the entire available width in the IVR system. The visual contents also included a display where each participant could see how many lives remained, a scoreboard and a time countdown starting at the total length of the game, 420 seconds. The spacecraft shot one projectile every 3 seconds towards the participant. A shot was considered a *hit* if the participant was in the same lane as a projectile when the it flew by, and a *miss* otherwise. The score was incremented by 1 every time they avoided a shot. If they got hit, one life was subtracted from the pool and the score was reset to zero. Participants were instructed to carry on with the game even if the life pool was empty, as long as the time countdown had not reached zero.

The projectiles travelled quickly enough so that a participant could not avoid it once it was shot ($7.5m/s$ and the participant was between $0.5m$ and $3.5m$ away). It was designed this way to encourage participants to try to develop a strategy based on prediction. The game score and the number of lives left were not relevant for the experiment and were not included in the

data analysis, but they proved to be very useful for keeping the participants engaged in the game.

2.2. The IVR system

The system used was a Cave-like virtual reality (VR) system similar to the one described in Cruz-Neira et al. (1993). The floor area was 3×3 meters and three walls 2.7 meters high. The images were rendered on all four surfaces, each one by a DLP projector with a resolution of 1440×1050 pixels with a refresh rate of 100Hz. The projectors were controlled from a cluster of 4 computers, each one equipped with an Nvidia Quadro FX 5600 graphics card. The participant wore light-weight Crystal Eyes shutter glasses synchronised with the rendering system to deliver stereoscopic images. The participant's head was tracked with an Intersense IS-900 tracking system to adjust the imagery from their perspective in real time. This system was chosen instead of a head-mounted display type because it allows the participant to wear just a pair of light weight shutter glasses and move around the space while still maintaining tracking. Furthermore, participants had to make sharp movements during the experiment, so shutter glasses and the Cave were safer than wearing a helmet that blocks out the sight to real world.

2.3. The RL design

The floor surface in the Cave was divided into 5 longitudinal lanes on the depth dimension, 60cms wide each. The current state of the RL machine was the lane the participant was in. This was computed at the time an action was taken. The action decided the lane from which the space craft would shoot and the projectile would travel along the lane towards the participant. In

summary, there were 5 lanes that the participant could be in and 5 possible positions the spacecraft could shoot from. All 5 actions were available on all 5 states. Thus the state-action map had dimensions 5×5 leading to 25 state-action possible permutations.

We used the on-line, off-policy algorithm Q(λ) as described in Sutton and Barto (1998) with the following RL parameters: learning rate $\alpha = 0.5$, discount rate $\gamma = 1$, and decay rate for eligibility traces $\lambda = 0.2$. ϵ represents the probability that the next action would be an exploratory one (choosing one randomly from all the possibles in the current state) or will exploit the best action. When $\epsilon = 1$, there is a 100% chance that the next action will be an exploration, and, $\epsilon = 0$ would mean that the agent will exploit the observed best action. In the non-random conditions, ϵ remained 1 for the first minute to encourage exploration, and then afterwards was decreased over time by -0.1 per step to progressively reduce the amount of exploration and increase the chance of using the accumulated experience, until it reached its minimum value, $\epsilon = 0.1$. In the Random condition, ϵ was 1 throughout the game. The reward obtained on each try was a discrete value that depended on the distance from the goal. If the participant was at the goal, then the reward would be 5. The reward would be then 1 less for each lane away from the target. The RL agent did not use the experience collected from previous participants, therefore it adapted for each participant individually.

2.4. Experimental design

The experimental conditions were Left, Right and Random. In the Left condition, the goal of the RL agent was to learn how to guide the participant to the leftmost lane. In Right, the target location was the rightmost

one. Random made the spacecraft shoot randomly throughout the game and did not use the experience collected at any time. Our decision to use the outermost areas of the Cave as the goal came after a pilot study, where we had asked volunteers about the place they felt safest. Most of them said that the centre was the safest, since staying there allowed them to move in any direction, thus having better options in the centre. Our hypothesis is related to whether we could override this feeling of safety and make them stay in a corner, thereby contradicting the most common response.

32 male participants were recruited among students at the university campus, all of them between 18 and 44-years-old with no significant differences between groups. Two participants had to be discarded due to technical problems recording the data. Participants were assigned to each experiment version alternately on arrival at the laboratory, with 10 participants in total in each group, in this between-group design. Once in the VR lab, they were instructed that the goal was to avoid the projectiles and they had to maximise the score displayed on the screen. No information about the RL agent's actual goal was given before the game. Each participant was paid £7 and it took about 25 minutes in total for each participant. This experiment was approved by the UCL Research Ethics Committee and participants gave written informed consent.

3. Results

The main response variable was the total reward obtained by the RL agent, as this measures how close a participant was from the goal. High reward values mean that a participant stayed closer to the goal and for longer

periods of time compared to others with lower scores. This is a single factor experiment, Version, and had three levels: The RL agent was trying to guide the participant to either the leftmost part of the Cave (Left), the rightmost part of the Cave (Right) or was shooting randomly throughout the game (Random). Our hypothesis was that the total reward obtained in Left and Right version would be similar and both be greater than in Random. Secondly, we expected the reward per action obtained in Left and Right during the game to increase over the time. This can also be defined as ϵ value being negatively correlated with the average reward per action obtained for these two versions.

One-way ANOVA was carried out for the response variable Reward on version, to test the null hypothesis of no difference in the mean rewards between the three conditions. This hypothesis is rejected with $F(2,27) = 116130$, $P = 0.0015$, $R^2 = 0.38$. Shapiro-Wilk test on the residual errors of the fit does not reject the assumption of normality ($P > 0.85$). Scheffe method overall confidence intervals for marginal differences show no significant difference between Right and Left (-60.09 to 213.69), a clear difference between Right and Random (-349.69 to -75.91) and support for difference between Left and Random (-272.89 to 0.89). Šidák multiple comparisons between groups provide further support for these results, having the 95% confidence interval values on the comparison between Random versus Left -270.5 to -1.5.

Concerning the progression of the rewards over the time, Figure 1 shows the average reward obtained in actions taken for each value of ϵ with the standard deviation represented by the whiskers on the bars. In early stages

of the game, when $\epsilon = 1.0$, the agent was only exploring and therefore the average reward obtained in Left and Right was similar to the reward obtained in Random ($Left = 1.91 \pm 0.74; Right = 2.18 \pm 0.82; Random = 1.86 \pm 0.87$). As ϵ started to decrease, the agent made greater use of the data collected and chose the actions that were more likely to return the highest reward. In the final stage of the game, for $\epsilon = 0.1$, the rewards obtained were Left (2.73 ± 0.61) and Right (2.9 ± 0.75).

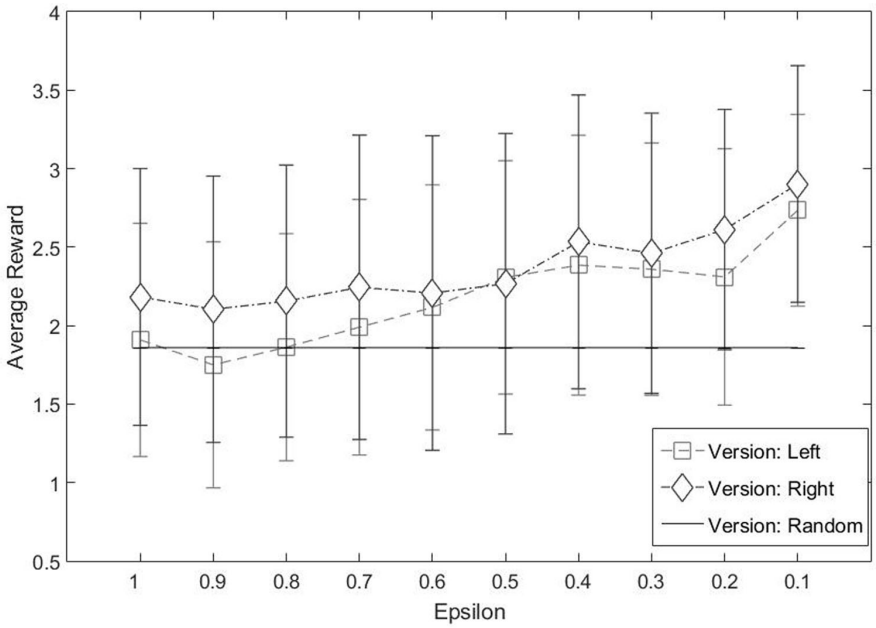
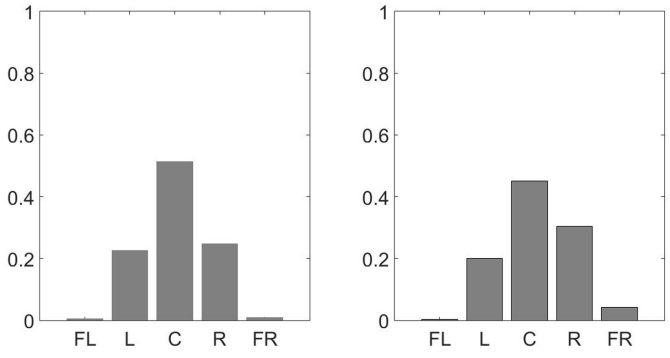


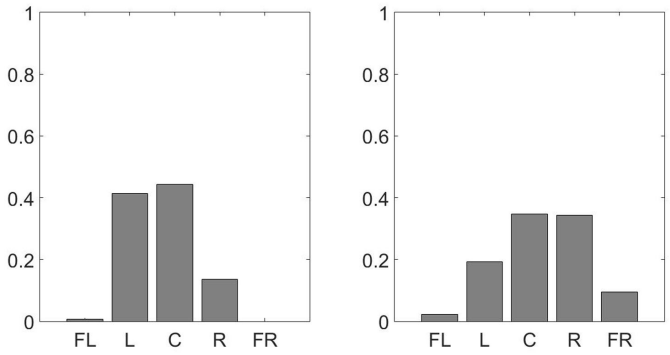
Figure 1: Mean and standard deviation for each ϵ grouped by experiment version.

The rewards obtained over the time can also be interpreted as the time spent in each area for each participant, since the reward obtained is inversely related to the distance from the goal area. The histograms of the distribution of time spent in each area for Left and Right version have roughly a symmetric

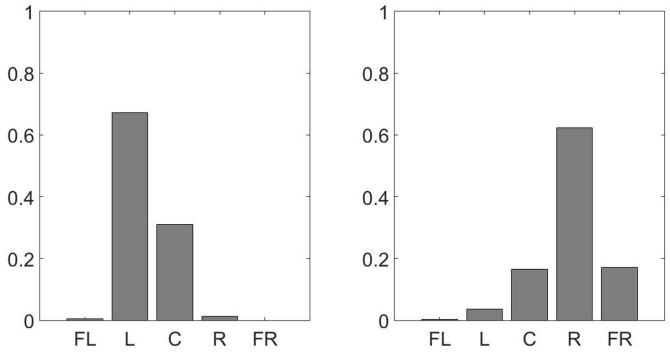
bell shape with the median on the centre value representing the middle lane in the Cave. The tendency of the participants to spend less time on the centre of the scenario as the ϵ decreased makes the histograms to skew towards the goal on each version. Figure 2 shows the histogram in three different stages, in the first stage of the experiment (Fig. 2a), half way through (Fig. 2b) and the last stage where the experience was used on 90% of the actions taken (Fig. 2c).



(a) $\epsilon = 1.0$



(b) $\epsilon = 0.5$



(c) $\epsilon = 0.1$

Figure 2: Percentages of time spent on each area for ϵ values 1.0, 0.5 and 0.1 . Left plots are from the Left version of the experiment, right plots are from the Right version. (FL=far left area, L=left, C=centre, R=right, FR=far right).

The significance levels of Kolmogorov-Smirnov tests to test the hypotheses that the Left and Right samples collected for each ϵ value are from the same distribution are shown in Table 1. The difference between Left and Right distribution functions for $\epsilon = 0.9$ and $\epsilon = 0.8$ is not significant. As ϵ decreases, the distribution functions for Left and Right rapidly move away from one another. Examining the evolution of the skewness as a measure of asymmetry in the distribution functions of the time spent on each area, both Left and Right start close to 0 for $\epsilon = 1.0$. As ϵ approaches the low values, the skewness values reach higher magnitudes. In the Left version, although not in constant progression, the level of skewness tends to increase over the time, while in Right the result is the opposite and move towards negative values (Fig. 3).

Epsilon	Left #samples	Left skewness	Right #samples	Right skewness	2-way KS test p-value
1.0	560	-0.06	559	0.10	0.16
0.9	292	-0.40	298	0.14	0.18
0.8	298	-0.07	295	0.19	0.51
0.7	298	-0.17	295	0.13	0.001
0.6	293	-0.04	299	0.14	< 0.001
0.5	596	0.35	592	-0.12	< 0.001
0.4	296	0.22	298	-0.20	< 0.001
0.3	593	0.18	592	-0.21	< 0.001
0.2	593	0.53	593	-0.47	< 0.001
0.1	294	0.24	297	-0.72	< 0.001

Table 1: Number of samples and skewness for each epsilon and experiment condition. KS test p-values show a progressive difference between Left and Right distributions as ϵ decreases.

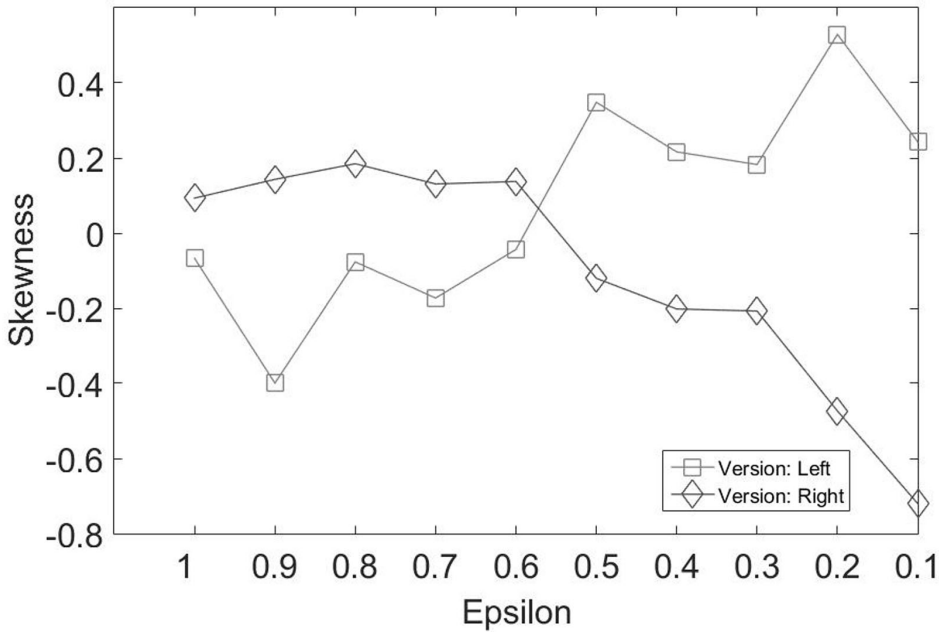


Figure 3: Skewness values of the histogram functions of the time spent on each area per ϵ for experiment versions Left and Right.

4. Discussion

The results show that the RL agent generally learned to guide participants towards the goal. In Left and Right conditions, the values obtained differ substantially from the ones in the version where the spacecraft was shooting randomly throughout the game, confirming our hypothesis. Despite the tendency that people moved towards the goal, the time spent at the goal area was still small. This is due to the fact that the goal was to make them stay at the corner and people thought it was a weak spot where the options to escape are reduced. Our goal was to override this natural feeling but the number of actions on each game might have needed to be higher to achieve

it. However, it is important to point out that there is a convergence towards the goal and, with a higher number of actions, it is likely that participants would have ended up staying at the corner for longer periods of time.

It is also interesting to note that we have used RL to influence the movements of the people. This is different from typical applications of RL, such as in board games or in Robotics. The target of the RL were the behaviours of the participants rather than those of virtual or robotic actors. Our experiment shows that RL can perform well in dynamic environments, since each person's strategy can be different from the rest based on their personality. Furthermore, a person might change his strategy over time. A RL agent is able to adjust its strategy by observing the outcome of the actions that takes.

Although the RL setup was 1D, the participant was unaware of this and was free to move anywhere in the 2D space of the Cave. However, in one sense this could be regarded as a replication study of the Kastanis and Slater (2012) study but applied in quite a different setup. Such replication studies are increasingly recognised to be important in science, since it is only through these that there is an ultimate validation of results. Additionally though, the present study has some important differences. In the Kastanis and Slater (2012) study emphasis was placed on the RL Agent eventually learning that the rules of proxemics operate in VR. The agent controlled an avatar that could go nearer or further away from the participant. Over time the agent learned that if it would go close to the participant then the participant would back away, moving her to the target position (a position unknown to the participant). In the new study the content of the situation is different in the sense that we rely on the RL agent learning that people will attempt

to dodge the virtual projectiles flying towards them, and that by targeting appropriately the participants can be constrained to particular areas of the space.

Some previous studies in RL have used techniques to mitigate the problem of having a large state-action space by adding a training session before the RL agent starts to solve the problem. In the context of our research, this could have also been applied to teach the RL agent how an average person behaves in our scenario and use it as a starting point. This would enable the RL agent to exploit this knowledge to make people move towards the goal in less time. Although it is possible to discover patterns of behaviour across participants, each individual has a different personality. This could lead the system to not converge to an optimal solution if the policy is based on a model created from other people. However, RL can be programmed to adjust its policy based on recent observations. In this experiment, the RL agent learned for each participant with no accumulated experience, but it was not difficult to observe common behaviour. Examples of this are the difficulties in making people stay in a corner, projectiles that were shot far away from the person were likely to make them stay idle, the tendency to move to the left when the projectile was shot very close to the right of the person, and vice versa. Nevertheless, the RL adapts individually to each person. It is does not assume that each one behaves in the same way. That is the power of this method, it is adaptive.

The applications that RL can have to influence on people's behaviour are not limited to IVEs. The same principles could be applied to other areas of human-computer interaction. For example, websites that want to maximise

the number of web pages visited or online stores in order to increase the sales by presenting different users with a range of different options that can dynamically change, and then learning over time the relationship (if any) between dynamic changes in content and the number of web pages read.

The design of the RL problem is critical and it is the key to a successful application using RL. The number of tries that the agent needs to complete is directly related to the number of possible state-actions pairs to ideally make sure that each pair has been tried a minimum number of times. But this is not always feasible due to the lack of time or because the environment changes too rapidly to test all of the pairs in an ideal frequency. In our experiment, the number of states and actions were reduced from the initial idea based on the observation of a pilot study with seven people, whose results have not been included in the analysis. The game length was also extended in order to increase the number of actions.

Virtual environments are built in the last analysis to influence participants – whether for entertainment, therapy, training, or some other goal. Usually how this influence operates is left to chance. We have shown how using RL it is possible to influence behaviour in a systematic way, that is adaptable to each participant. Clearly more complex examples need to be studied for future applications. In the future, we aim to include a RL agent to the scenario presented in Slater et al. (2013) where a participant faced a violent emergency between two virtual characters and had to decide whether to intervene in order to stop them arguing or step back and do nothing about it. In this upcoming scenario, the RL agent will make the virtual characters perform certain actions to learn how the likelihood of intervention can be

maximised.

Acknowledgements

This research was funded by UK EPSRC Project EP/F032420/1 "Visual and Behavioural Fidelity of Virtual Humans with Applications to Bystander Intervention in Violent Emergencies".

References

Cruz-Neira, C., Sandin, D. J., DeFanti, T. A., 1993. Surround-screen projection-based virtual reality: the design and implementation of the CAVE. In: SIGGRAPH '93 Proceedings of the 20th annual conference on Computer graphics and interactive techniques. pp. 135–142.
URL http://dl.acm.org/citation.cfm?id=166134

Garcia-Palacios, A., Hoffman, H., Carlin, A., Furness, T. a., Botella, C., 2002. Virtual reality in the treatment of spider phobia: A controlled study. Behaviour Research and Therapy 40, 983–993.

Hall, E. T., 1966. The Hidden Dimension. Anchor.

Held, R., Durlach, N., 1991. Telepresence, time delay and adaptation. In: Pictorial Communication in Real and Virtual Environments. Taylor & Francis, Inc., Bristol, PA, USA, pp. 232–246.
URL http://books.google.com/books?hl=en&lr=&id=8c6PClwCymcC&oi=fnd&pg=PA232&dq="

Hoffmann, G., Jang, J. S., Tomlin, C. J., 2005. Multi-Agent X4-Flyer Testbed Control Design: Integral Sliding Mode vs. Reinforcement Learning. International Conference on Intelligent Robots and Systems, 468–473.

IJsselsteijn, W., de Ridder, H., Freeman, J., Avons, S. E., Bouwhuis, D., 2001. Effects of stereoscopic presentation, image motion, and screen size on subjective and objective corroborative measures of presence. Presence: Teleoperators & Virtual Environments 10 (3), 298–311.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6790855

Kaelbling, L. P., Littman, M. L., Moore, A. W., 1996. Reinforcement Learning: A Survey. Journal of Artificial Intelligence Research 4, 237–285.
URL http://arxiv.org/abs/cs/9605103

Kastanis, I., Slater, M., 2012. Reinforcement Learning Utilizes Proxemics : An Avatar Learns to Manipulate the Position of People in Immersive Virtual Reality. Transactions on Applied Perception 9 (1).
URL http://dl.acm.org/citation.cfm?id=2134206

Kober, J., Bagnell, J. A., Peters, J., 2012. Reinforcement Learning in Robotics : A Survey. In: Reinforcement Learning. Springer Berlin Heidelberg, pp. 579–610.
URL http://link.springer.com/chapter/10.1007/978-3-642-27645-3_18

Kolter, J. Z., Ng, A. Y., 2009. Policy Search via the Signed Derivative. In: Robotics: science and systems.
URL http://roboticsproceedings.org/rss05/p27.pdf

Kormushev, P., Calinon, S., Caldwell, D., jul 2013. Reinforcement Learning in Robotics: Applications and Real-World Challenges. Robotics 2 (3), 122–148.
URL http://www.mdpi.com/2218-6581/2/3/122/

Lee, J., Lee, K. H., mar 2006. Precomputing avatar behavior from human motion data. Graphical Models 68 (2), 158–174.
URL http://linkinghub.elsevier.com/retrieve/pii/S1524070305000275
http://www.sciencedirect.com/science/article/pii/S1524070305000275

Lin, J.-W., Duh, H., Parker, D., Abi-Rached, H., Furness, T., 2002. Effects of field of view on presence, enjoyment, memory, and simulator sickness in a virtual environment. In: Proceedings IEEE Virtual Reality 2002. IEEE Comput. Soc, pp. 164–171.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=996519

Marthi, B., Russell, S. J., Latham, D., 2005. Writing Stratagus-playing Agents in Concurrent ALisp. In: Proceedings of the IJCAI-05 Workshop on Reasoning, Representation, and Learning in Computer Games. Citeseer, p. 67.
URL http://www.cs.auckland.ac.nz/compsci777s2c/ijcai05.pdf#page=71

Meehan, M., Razzaque, S., Whitton, M. C., Brooks Jr., F. P., 2003. Effect of Latency on Presence in Stressful Virtual Environments. In: IEEE Virtual Reality, Proceedings. IEEE, pp. 141–148.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1191132

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D., 2015. Human-level control through deep reinforcement learning. Nature 518 (7540), 529–533.
URL http://dx.doi.org/10.1038/nature14236

Ng, A. Y., Kim, H. J., Jordan, M. I., Sastry, S., 2004. Autonomous helicopter flight via Reinforcement Learning. Advances in Neural Information Processing Systems 16 16, 363–372.
URL http://www.springerlink.com/index/w3618557nl185574.pdf

Noe, A., 2004. Action in Perception. MIT Press.

Pertaub, D.-P., Slater, M., Barker, C., 2002. An Experiment on Public Speaking Anxiety in Response to Three Different Types or Virtual Audience. Presence: Teleoperators & Virtual Environments 11 (1), 68–78.
URL http://www.mitpressjournals.org/doi/abs/10.1162/105474602317343668

Rothbaum, B. O., Hodges, L., Alarcon, R., Ready, D., Shahar, F., Graap, K., Pair, J., Hebert, P., Gotz, D., Wills, B., Baltzell, D., 1999. Virtual reality exposure therapy for PTSD Vietnam Veterans: a case study. Journal of traumatic stress 12 (2), 263–271.

Samuel, A. L., 1959. Some Studies in Machine Learning Using the Game of Checkers. IBM Journal of research and development 44 (1.2), 206–226.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5389202
http://www.research.ibm.com/journal/rd/033/ibmrd0303B.pdf

Samuel, A. L., 1967. Some studies in machine learning using the game of checkers. II - Recent progress. IBM Journal of Research and Development 6 (11), 601–617.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5391906

Sanchez-Vives, M. V., Slater, M., apr 2005. From Presence to Consciousness Through Virtual Reality. Nature Reviews Neuroscience 6 (4), 332–339.

URL http://www.ncbi.nlm.nih.gov/pubmed/15803164
http://www.nature.com/nrn/journal/v6/n4/abs/nrn1651.html

Slater, M., Rovira, A., Southern, R., Swapp, D., 2013. Bystander responses to a violent incident in an immersive virtual environment. PloS one 8 (1), 13.
URL http://dx.plos.org/10.1371/journal.pone.0052766

Slater, M., Usoh, M., Steed, A., 1994. Depth of presence in virtual environments. Presence: Teleoperators and Virtual Environments 3 (2), 130–144.
URL http://s3.amazonaws.com/publicationslist.org/data/melslater/ref-24/depth of presence.pdf

Spronck, P., Sprinkhuizen-kuyper, I., Postma, E., 2003. Online Adaptation of Game Opponent AI in Simulation and in Practice. In: Proceedings of the 4th International Conference on Intelligent Games and Simulation. pp. 93–100.
URL http://www.dcc.ru.nl/ idak/publications/papers/SpronckGAMEON2003.pdf

Sutton, R. S., Barto, A. G., jan 1998. Reinforcement Learning: An Introduction. Vol. 104. MIT Press.
URL http://books.google.com/books?hl=en&lr=&id=CAFR6IBF4xYC&oi=fnd&pg=PA3&dq=Re

Treuille, A., Lee, Y., Popović, Z., 2007. Near-optimal character animation with continuous control. ACM Transactions on Graphics 26 (3).
URL http://portal.acm.org/citation.cfm?doid=1275808.1276386 http://dl.acm.org/citation.cfm?id=1276386

Vigorito, C. M., 2007. Distributed path planning for mobile robots using a swarm of interacting reinforcement learners. In: Durfee, E. H., Yokoo, M., Huhns, M. N., Shehory, O. (Eds.), Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems - AAMAS '07. Vol. 5. ACM Press, New York, New York, USA, p. 1.
URL http://doi.acm.org/10.1145/1329125.1329273 http://portal.acm.org/citation.cfm?doid=1329125.1329273

Wiering, M., van Otterlo, M., 2012. Reinforcement Learning: State-of-the-art. Springer.
URL http://link.springer.com/content/pdf/10.1007/978-3-642-20536-1.pdf

Figure Captions

Figure 1: Mean and standard deviation for each ϵ grouped by experiment version.

Figure 2: Percentages of time spent on each area for ϵ values 1.0, 0.5 and 0.1 . Left plots are from the Left version of the experiment, right plots are from the Right version. (FL=far left area, L=left, C=centre, R=right, FR=far right). Figure 2.(a): $\epsilon = 1.0$

Figure 2.(b): $\epsilon = 0.5$

Figure 2.(c): $\epsilon = 0.1$

Figure 3: Skewness values of the histogram functions of the time spent on each area per ϵ for experiment versions Left and Right.