

Serious Games & Human Reliability. The Use of Game-Engine-Based Simulator Data for Studies of Evacuation Under Toxic Cloud Scenario.

Marcos V. P. Andrade^a, Caio B. Souto Maior^{a,*}, Erika O. Silva^a, Márcio C. Moura^a and Isis D. Lins^a

^a Center for Risk Analysis and Environmental Modeling, Department of Production Engineering,
Federal University of Pernambuco, Recife, Brazil

Abstract: The rapidly evolving technology turned industrial process automation into tangible reality. As man-made activities are still undoubtedly necessary, especially in complex and cognitive tasks, human behavior and performance have become a major concern in almost every economic activity. Human factors are quite often pointed, direct or indirectly, as one of the main accident causes in areas such as aviation, chemical industry, nuclear power generation and water supply among many others. The present work proposes to discuss the use of game engines in the construction of virtual environments for Human Reliability Assessment (HRA) studies. Two well known facts motivate this work: the lack of good datasets for HRA estimations, and the shortage of funding for scientific research, generating small teams with poor interdisciplinarity. This paper discusses how to port real-world scenarios to virtual environments aiming to extract useful data for HRA and the use of in game Analytics, from Serious Games (SG) gameplay sessions, under the small teams' perspective. The work discusses how to adapt scenarios, create analytics generation setups and shows which data types can be obtained. Also, a real scenario accident of an evacuation and toxic cloud release is modeled and analyzed to study the proposed methodology feasibility.

Keywords: Human Reliability, Game Engines, Analytics, Simulators.

1. INTRODUCTION

The concern about human behavior and performance have become a major issue in almost every economic activity these days. The intuitive notion of human factors as one of the main causes of accidents, is being confirmed by studies in areas such as aviation [1], chemical industry [2], nuclear power generation, [3] and water supply [4], among others, turning what was intuitive into a scientific "de facto".

The above-mentioned studies discuss the importance in analyzing and infer human performance levels and error probabilities in socio-technical systems. However, the scarcity of available data, specifically human error data, represents a relevant obstacle in the Human Reliability Assessment (HRA) field [5] – [7].

In this context, recent works bring interesting alternatives to overcome this problem. Refs. [8], [9] present digitally created virtual environments to simulate an evacuation scenario with the resulting data being used to successfully feed a Bayesian Network. Then, the resulting data is used to calculate the conditional probabilities and the likelihood of success for the tasks studied and under predefined circumstances. In another experiment, Ref. [10] uses a driving simulator to study sleepiness at the steering wheel, also showing good prospect in the use of a digital double to avoid risky situations, in accordance with conclusions achieve by Ref. [11].

The present work aims to explore how to create scenarios in a GE-based simulator so that they can be used to feed a human reliability analysis. The approach has already been discussed in the literature in cases of virtual environments generating data to HRA studies or to validate concepts [1]–[3], but about the subject of how the Virtual Environment (VE) was created, very few, or no info at all is given. At the

*Corresponding author: Caio Bezerra Souto Maior (caiomaior@hotmail.com)

best of authors' knowledge, Ref [4] is the only that is a bit clearer on the subject of environment creation though it is not tailored for HRA.

For instance, in the Gas and Oil area, Visiongain's report on the oil and gas training market for the period 2011-2021 provides a growth forecast for investment in training based on dynamic platforms such as Virtual Reality (VR) and Serious Games (SG) from moderate to high [5]. This large demand for rich content training denotes a positive environment to undertake HRA studies with virtual environments aggregate data, with scientific gains, optimizing costs and efforts expended.

In fact, simulators use has been in a growing trend, and many of them are based on virtual reality (VR), which reduces costs and increases training efficiency. There are several training solutions using the so-called Game Engines (GE), like the ones created by Immerse Inc. (2017).

These tools are now at easy reach, depicting a context where a framework to assist the creation of immersive virtual environments (VE) to study human reactions, performance, and behavior, becomes a must-have solution, deserving comprehensive studies.

Therefore, it is clear the relevance of researching alternatives for systematic data collection beyond traditional post-training assessments, and research or accident reports [12]. Then, the present work proposes to discuss the use of Game Engines (GE) in the construction of virtual environments for Human Reliability Assessment (HRA) studies.

1. DEFINING GAME ENGINES (GE)

Game engines consist of a set of game design tools grouped into a unique computational environment. Roughly speaking, GE could be compared to text editors, where all the necessary tools are present in the software environment (e.g., printing modules, spelling, and formatting). In game engines, the tools needed from the conception to the final output of an application are implemented and designed to be reusable and project-independent.

The various embedded features allow the creation of electronic games, simulations, or any application that requires real-time graphics, simulations and user interaction. The contemporary GE have layers of hardware abstraction that allow the creation of applications for all other devices on a single platform. Besides that the most common features found in game engines are [7], [8]:

- Graphic engine to generate two-dimensional, three-dimensional and stereoscopic graphics;
- Physics engine for simulations;
- Artificial intelligence (AI) engine for character behaviors;
- Interface for programming languages and scripts like C ++, C#, JavaScript and others;
- Multi-player network management;
- Virtualizers to simulate the various delivery platforms for prototyping.

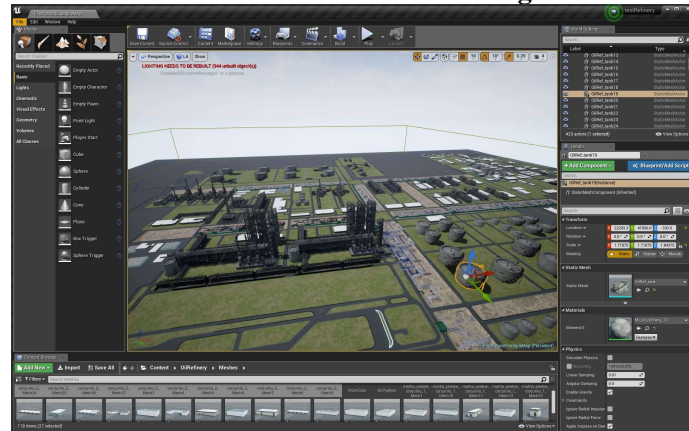
From the solutions on the market (or about to be), the following GE are noteworthy: Unity Engine www.unity3d.com, Cry Engine www.cryengine.com [9], Ogre www.ogre3d.org. The giant Amazon is launching Lumberyard aws.amazon.com/pt/lumberyard/, its own proprietary engine, and Autodesk, another giant in computer graphics, launched Stingray www.autodesk.com/products/stingray/overview.

In the present article, we use the Unreal Game Engine www.unrealengine.com [10], which is free for academic purposes, has outstanding graphic quality and a more user friendly approach. The software has no working limitations and all versions are available for download at no charge.

The Unreal, among the main features, has a visual programming interface called Blueprints, which permits almost every aspect of the environment to be programmed and controlled without any written code. This GE has a huge user base and excellent interoperability with the major 3D software packages

via FBX file pipeline. **Erro! Fonte de referência não encontrada.** shows an example of the created scenarios inside the Unreal GE with an example of its corresponding Blueprint script.

Figure 1 -Scenario overview in the Unreal Game Engine. Source: the research



2. RELATED WORK

Ref. [11] presents an interesting experiment for firefighters training using data from an external Computer Fluid Simulation software applied inside a Game Engine. The experiment has many simplifications and the dynamic process simulator data has a link that works only inwards the GE. Being fed at the session start so player reactions did not generate feedback for the process simulator to reprocess the scenario logic.

The host system is quite modest for the task and authors mention some performance issues, mainly in the form of poor performance in frames per second, which is a common benchmark for game levels. This means that the host system can not recalculate the changes in the scenario in a rate fast enough to keep all motion fluid in the simulation. Pictures from paper show some 3D models with unnecessary details like a car's wheels with excessive polygons. The system uses VR glasses for immersion and a joystick and motion trackers for interactions. The overall assessment of the application by professional firefighters indicates the VE for inexperienced novices and suggested some game mechanics improve efficiency.

Despite simplifications that players actions do not affect the fire simulation, which represents a huge gap in training firefighterst, it is an interesting initiative for desirable use of more powerful hardware to reduce the bottlenecks in system performance.

With good ideas on 3D modeling optimizations and a good workflow for 3D modeling software and GE, the work in Ref. [12] brings a Virtual Reality (VR) simulator of a human heart to enhance anatomy classes which, according to the author, have been suffering from various difficulties in using cadaveric materials. The article has many suggestions on user interaction and interface, done in Unity game engine, with an intermediate step at 3DS Max. Recommendations are given about system performance and models optimization to guarantee minimal performance during the interactions. The same system has also a simplified version aimed at clarifying heart diseases concepts to patients.

In an ingenious way to use empiric simulator data to do HRA evacuation studies, the reference [1] presents a virtual training environment called AVERT in which it develops route data and routing decisions in offshore installation evacuation. It concludes that, although the travel times are not 100% realistic, the interrelationships between virtual performances are very reliable to what is expected in a real environment. The variables studied are training, facility complexity and visibility and the experiment can have enough data to establish the conditional probabilities for a Bayesian Belief Network (BBN) [1], also making it possible to validate values for the Performance Shaping Factor (PSF) [1]. The research uses a proprietary virtual environment and doesn't talk about how the metrics are collected and

as it is based on a proprietary system the access problem remains since there's no free and ubiquitous access to the AVERT system.

In Ref. [13] a serious game simulation is used in emergency preparedness for terrorist attacks on a train station. The experiment consisted of a simulated terrorist attack a passenger train. As the subsequent events unfold, the player must make decisions to survive the attack. Every wrong choice that leads to damage or death is pointed and explained in dialog boxes that appear on screen (Figure 2)

Figure 2: Death cause informed on-screen information, and security recommendations to avoid that in similar scenarios. Source:(CHITTARO; SIONI, 2015)

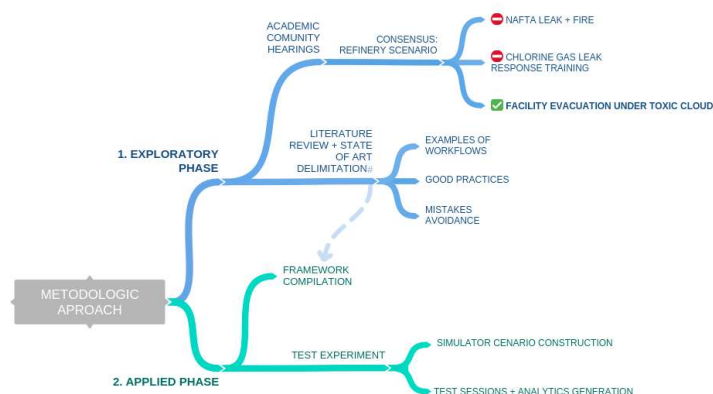


The scenario runs on a 30-inch computer screen and interaction is done via Nintendo Wii controllers and, despite being mentioned the use of a GE, no further specification is given. Some stimuli are applied to the player in order give the dimension of the threat involved and the seriousness of the consequences in a way similar to game mechanics. The Heart Rate (HR) and Skin Conductance Level (SCL) of the players are logged via an Electrodermal Activity (EDA) sensor.

3. METHODOLOGY

In this paper, we propose a systematic approach for generating GE based simulator scenarios (Figure 3). For the present work, we assume that the data generated will feed a Bayesian Network as in works like [7], [8], [22] – [24].

Figure 3: Schematic of methodologic approach. Source: The research.



The proposed experiment created a Virtual Environment (VE) for a toxic cloud release, an Analytics setup to gather and record player interactions and other aggregate data, as well as, the conversions to useable formats from the raw data stream outputted by the GE. The conceptual setup for the Bayesian Network is presented with all arcs and nodes, but the final calculations will be left for future works.

The process of creating a virtual simulator for a specific HRA scenario study, comprehends a series adaptations and scope narrowing choices, the related works research brought examples of good practices and the game development literature also had some interesting approaches. For porting such scenarios to virtual environments some good practices list was compiled with a brief description of each one. [14]–[16].

Some of these steps fall at the learned lessons category and where created based to address problems occurred during experiment construction. The final step, which is not conducted in the present work, was kept here to illustrate all steps in the technique.

Then, the steps are:

- Scenario Definition – In this step, all the events to happen in the scenario are physically and logically described with all branches of possible actions covered;
- Choice of the immersion style: will it be Virtual or Augmented Reality or a mix of both.
- Interactions metaphors definition: how does the subject interact in the scenario – a realistic decision must be made taking into consideration the capabilities of the team;
- 3D models building, and/or CAD models conversion – Most of the design in the engineering process today is done in 3D, so reusing these models may streamline the scenario creation.
- It is important to have in mind that some adaptations and cleanup in the 3D meshes will be necessary;
- Objects/Actors placement on the scenario: After all the so-called assets are created, they must be “physically” placed in the virtual environment. Parked cars, trees, and other elements must be positioned in the virtual environment;
- Intra-game programming of the working laws in the scenarios - Game engines have programming languages to enable environment reactions, logic processing, simulations tuning, and any kind of customization needed;
- Event definition and work variables – What will happen as the user/player progresses through the simulation, how will the relevant data be stored. See **Erro! Fonte de referência não encontrada.** and **Erro! Fonte de referência não encontrada.**;
- Quantitative approach definition: what will method be used for HRA? In this article, we considered the used scenario is part of an HRA performed by the Phoenix technique [17];
- Definition of the metrics to be collected during gameplay: In this step, all data that will be generated during the game session is defined, e.g. reaction times, actions taken and any other event that is of interest for the study (**Erro! Fonte de referência não encontrada.**);
- Database taxonomy definition: how will the database records be composed of, what context variables will be stored or any other definition regarding data recording must be previously defined;
- Game mechanics policy definition (how much interference or stimulus must be applied, what will simplifications be done. Sometimes some adaptations or some stimulus may be applied intra virtual environment to achieve determined ends, e.g., will be any time constraints? Will user results be visible to other users? Will some manipulations of components be simplified? There will be a right or wrong feedback in real-time for the user.
- Test Gameplay Sessions: Users will interact with the environment to generate data from their actions inside the game;
- Post session debriefing for user impressions (recorded on video or audio only when possible): Is good to listen to the users immediately after a simulated session to catch impressions and valuable insights permitting to improve the simulator logic, and understand user interpretation in general;
- Benchmark with prior works when possible: Compare with pre-existent data about the studied events, if available.

3.1. Scenario Definitions

A refinery was created on the simulator to allow analysis of an evacuation process without having to do real large-scale simulated exercises, which despite being very important are difficult to manage and

expensive. Based on visual research of refineries models, the digital prototype was created in scale inside Sketchup with the streets, sidewalks and blocks for the administration buildings were drawn in scale.

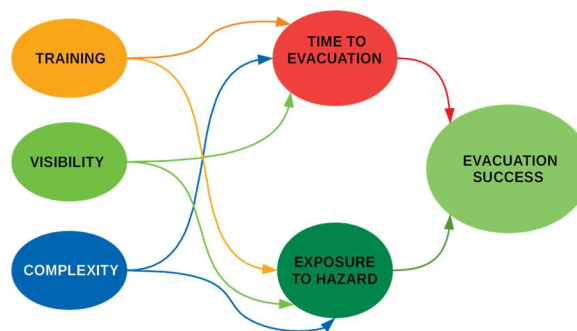
The major concern in the case was to maintain similarity with a refinery at ground level so players walking the streets will be somewhat familiar with the surroundings. Due to the modular nature of Game Engines (GE), any geometry placed on the terrain could be promptly replaced keeping very simple the process of updating and modifying geometries in the scenario (Figure 4).

Figure 4: General view of the modeled Refinery, inside the Game Engine. Source: The author.



The scenario consisted of an evacuation of the plant due to a leak in the Sour Water Treatment that generated a toxic cloud of Hydroxy Sulfide (H_2S). It is based on the scenario used by [18] a path optimization work. The main objective was to estimate of the possibility of successfully evacuate the facility in a toxic cloud situation under various conditions correlated with the random variables in a causal network as represented in (Figure 5).

Figure 5: The Bayesian Belief Network (BBN) for the evacuation procedures. The first three from the left are random variables the two middle and the right ones are the dependent ones. Source: the research.

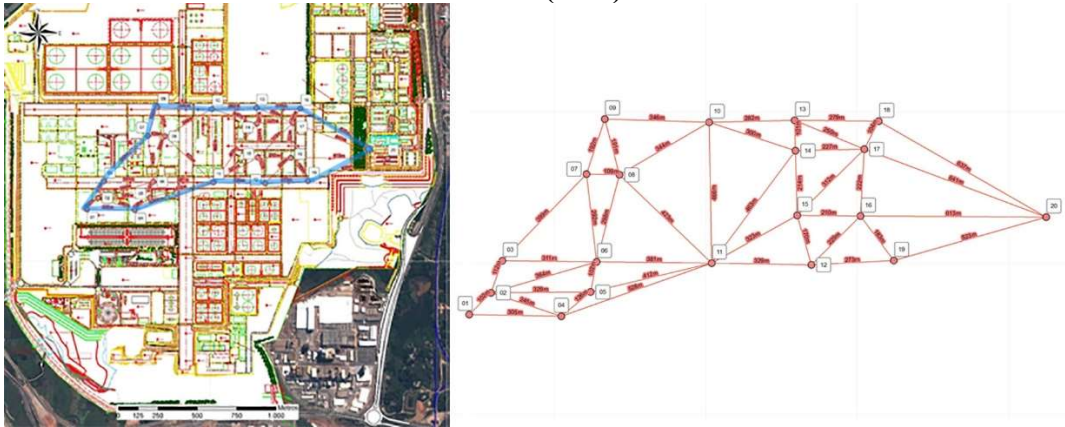


A 20-point graph (Figure 6) of all possible routes, from the initial point to the final one, was used, with landmarks placed on the terrain to be used as checkpoints for the player leaving the area. The points were not visible to the player, but each checkpoint has a control area that when trespassed by the player, records his presence, a timecode of when the sensor was triggered and the cloud concentration on the area as a way to calculate the absorbed dose on route.

For auxiliary data, an extensive mesh of 176 additional smart-sensors (Figure 7) was created so the movements of a player during an evacuation could be tracked for additional studies. Every sensor also worked as a stopwatch and had internal metadata with the concentration of H_2S at its location.

For computing the time spent in the whole exercise, a universal time counter is started at the begin of the session and a lap is counted every time a checkpoint is reached. For the exercise, an average speed of 6 km/h was assumed for walking with two upper grades for fast walking and running permitted. All the results were grouped into two categories, above and below the benchmark time.

Figure 6: Refinery plant with all the 20 checkpoints placed on the terrain. Source: Oliveira da Silva (2017)



For the visibility variable a copy of the scenario was built with a night lighting arrangement, so was possible run the same scenario in day and night situations. No auxiliary light is provided to the player at the night session.

Figure 7: Smart sensors placement in all plant, superimposed on the original grid. Source: The Research.



For the complexity variable, two routes are set and the player must do both sessions with two complexity levels to complete the exercise. The first scenario was easier to complete, and the second was a bit more complex with a part of it passing through some equipment to slow down the player progression. A total of three scenarios (Table 1) were simulated: 1) day with low complexity; 2) night time with low complexity; 3) with daytime and high complexity. All walking or running takes places in the streets and sidewalks of the plant. No interaction with any type of equipment is allowed so the player really must leave the place as quickly as possible avoiding the risky areas.

At the beginning of the simulation, a brief description of the event is presented to the player and the suggested route is presented as a combination of the checkpoints. At the end of the route, all concentrations and times are used to calculate total risk of dying on route based on Probit equations.

The Probit equations are used to estimate the potential lethality of a toxic load, they are basically a function of concentration and exposure time [19]. This equation estimates the probability of death of an individual exposed to a substance, with a specific concentration for certain period of time [18], [20]. The Probit equation have are constants in which its values dependent on the substance in study, concentration and exposure time [19], [18], [20]. More details about Probit equation are given in section 3.3.

Table 1: Scenario settings for data generation. Source: The research.

Scenario #	Visibility	Complexity
scn1	day	low
scn2	night	low
scn3	day	high

A delimited area, where the consequences of toxic cloud poisoning could be lethal, is invisibly defined in the scenario. If the player passes through a smart sensor, with a non-zero concentration on it, a biohazard event is logged with the concentration on that point being logged by the Analytics engine.

In these kinds of projects normally is expected to use geometry from CAD models with minor modifications. The final Refinery model was not 100% faithful to any plant but general layout was made from visual references and is similar enough to permit extend conclusions drawn from the model to a real facility.

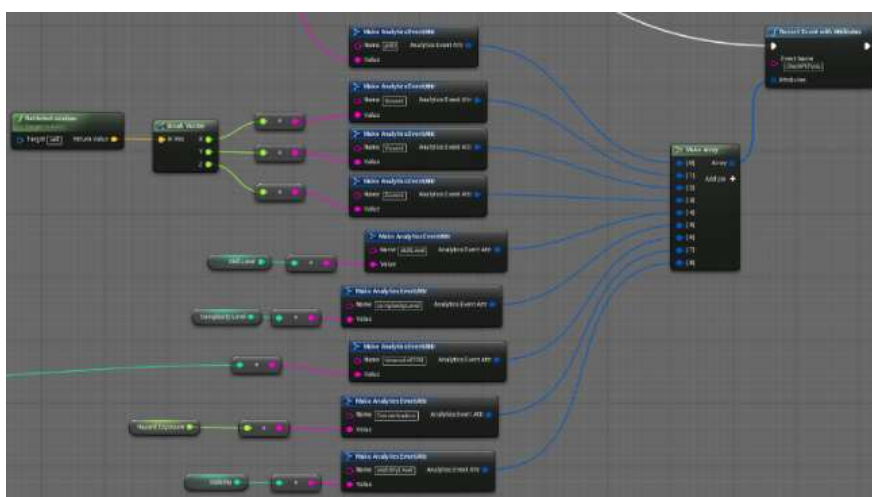
3.2. Analytics setup and data handling

Two paths of output were simultaneously used for the data generated on simulator sessions:

- Via Google Analytics plugin, for real-time remote monitoring of user actions on scenario via Events and custom metrics and dimensions via the internet. This feature was successfully implemented and tested.
- Local disk file recording, where database files containing records of all passages through the smart sensors with all study variables logged in each passage and their state throughout the role route. The files are written in *JSON* database exchange format, that can be easily read and converted to spreadsheets for posterior processing (Figure 8 and Table 2).

For each smart sensor, a routine to output its coordinates and input it at Aloha® (Areal Locations of Hazardous Atmospheres) software for estimating the gas concentration in each point was created. Then, a CSV file would receive all concentrations in each point and write them to the smart sensors.

Figure 8: Blueprint implementation of Analytics recording on local disk. Source: The research



This data will be present in the final report enabling to estimate the hazard exposition on the route for every session. This coupling Aloha – Unreal engine was done via datatables, an intrinsic way for the Unreal GE to handle external data sources, permitting the values to be updated for changing the characteristics of each scenario in an automated way.

Table 2: Partial extract of raw data outputted from GE Analytics plugin. Source: The research.

ID #	TRAINING	VISIBILITY	ORDER IN SESSION	COMPLEXITY	TIME (MIN)	TIME STATUS	EXPOSURE STATUS	ABSORBED DOSE	STATUS
1	MINIMAL	DAY	1st	SIMPLE	17.06667	WORSE	BELOW	19,235.17	SUCCEED
1	MINIMAL	DAY	2nd	COMPLEX	31.63333	WORSE	BELOW	36,342.99	SUCCEED
1	MINIMAL	NIGHT	3rd	SIMPLE	20.71667	WORSE	BELOW	17,275.18	SUCCEED
2	EXPANDED	DAY	1st	SIMPLE	9.78333	BETTER	BELOW	1,752.55	SUCCEED
2	EXPANDED	DAY	2nd	COMPLEX	12.55000	BETTER	ABOVE	390,869.48	FAILED
2	EXPANDED	NIGHT	3rd	SIMPLE	9.63333	BETTER	ABOVE	390,448.75	FAILED
3	MINIMAL	DAY	1st	SIMPLE	62.08333	WORSE	ABOVE	2,120,169.71	FAILED
14	MINIMAL	DAY	2nd	SIMPLE	27.41667	WORSE	BELOW	95,174.32	SUCCEED
14	MINIMAL	DAY	3rd	COMPLEX	15.40000	BETTER	BELOW	26,914.42	SUCCEED
14	MINIMAL	NIGHT	1st	SIMPLE	29.33333	WORSE	ABOVE	1,629,338.75	FAILED
15	EXPANDED	DAY	1st	SIMPLE	16.45000	BETTER	ABOVE	2,276,660.83	FAILED
15	EXPANDED	DAY	2nd	COMPLEX	15.96667	BETTER	BELOW	48,593.31	SUCCEED
15	EXPANDED	NIGHT	3rd	SIMPLE	13.91667	BETTER	BELOW		SUCCEED
REFINERY STAFF MEMBER BENCHMARK TIME								SESSIONS SUCCESS %	39.53%
								AVG MOG Exposure*	3.46E-06
								AVG MOG Time*	41.54000

3.3. Calculating Hazard exposition on route

The hazard on the route was calculated via Probit equations – Eq. (1) -, that are used to estimate the probability of death for an individual exposed to the hazardous substance, based on the substance concentration and the individual exposure time to it:

$$Pr = a + b \cdot \ln(C^n \cdot t_e) \quad (1)$$

where:

Pr – Probit equations, a Gaussian distribution with mean 5 and standard deviation 1.

a, b, n - Constants related to substance toxicity.

C - Concentration in mg/m³.

t_e - Exposure time in minutes.

There are several possible values for the constants in this equation when dealing with H₂S and this work adopted ones used in contemporary references such as [18], [21]:

$$a = -11.5, b = 1 \text{ and } n = 1.9$$

The set of values has been used to estimate the dose absorbed on the route. Keep the same constants as other papers (e.g. [18], [21]) enables a more coherent comparison of the risks on routes and times, whether using the optimized ones or done by users on the simulator. a more coherent comparison.

The analytics report is a JSON file outputted straight from the GE and carries context variables like visibility, training level of the user, unique session identifier and anonymous user tagId. There was also quantitative data of the simulator session containing for all activated sensors on route, position, activation timecode and Hydroxy Sulfide concentration as can be seen in Table 2 Table 2

From this report, it was quite straightforward to calculate the dose absorbed on route via a simple excel spreadsheet (Table 3) converted from the JSON report and estimate the hazard exposition on route [18]. It was also possible to calculate the hazard exposition inside GE, but the external spreadsheet was preferred for being more streamlined and permitted more flexibility.

Table 3: Values calculated for hazard exposition from analytics report only the two rightmost columns were calculated on a spreadsheet. Source: The research.

attributes/6/name	ributes/6/va	tributes/7/nam	tributes/7/va	tributes/8/nam	tributes/8/va	Concentr. mg/m3	Exp. Time (min)	DOSE PER SEGMENT	
timecodeX100	1411	Concentration	21.690001	visibilityLevel	1	31.258637	0.00000	-	
timecodeX100	1678	Concentration	78.610001	visibilityLevel	1	113.289135	0.44500	3,558.92	DOSE ON ROUTE
timecodeX100	2826	Concentration	648.729980	visibilityLevel	1	934.919954	1.91333	843,843.78	1,850,224.80
timecodeX100	3940	Concentration	593.590027	visibilityLevel	1	855.454778	1.85667	691,684.98	PROBIT EQ RESULT
timecodeX100	5075	Concentration	198.259995	visibilityLevel	1	285.723230	1.89167	87,728.59	2.930817703
timecodeX100	6082	Concentration	54.970001	visibilityLevel	1	79.220249	1.67833	6,802.48	CASUALTY PROBABILITY
timecodeX100	6950	Concentration	16.900000	visibilityLevel	1	24.355506	1.44667	623.60	1.93%
timecodeX100	8285	Concentration	248.410004	visibilityLevel	1	357.997128	2.22500	158,380.29	
timecodeX100	8518	Concentration	0.000000	visibilityLevel	1	0.000000	0.38833	-	
timecodeX100	9259	Concentration	148.970001	visibilityLevel	1	214.688747	1.23500	33,273.97	
timecodeX100	10734	Concentration	0.000000	visibilityLevel	1	0.000000	2.45833	-	
timecodeX100	12075	Concentration	0.000000	visibilityLevel	1	0.000000	2.23500	-	
timecodeX100	12496	Concentration	0.000000	visibilityLevel	1	0.000000	0.70167	-	
timecodeX100	13083	Concentration	0.000000	visibilityLevel	1	0.000000	0.97833	-	
timecodeX100	16092	Concentration	4.040000	visibilityLevel	1	5.822263	5.01500	142.54	
timecodeX100	16773	Concentration	9.580000	visibilityLevel	1	13.806258	1.13500	166.39	
timecodeX100	17901	Concentration	72.260002	visibilityLevel	1	104.137807	1.88000	12,811.91	
timecodeX100	21164	Concentration	38.480000	visibilityLevel	1	55.455615	5.43833	11,193.41	
timecodeX100	22612	Concentration	0.000000	visibilityLevel	1	0.000000	2.41333	-	
timecodeX100	23994	Concentration	1.790000	visibilityLevel	1	2.579666	2.30333	13.94	

3.4. Data classification for Bayesian Network Creation

As said, the present work makes a straight interface with [18] in the sense that the obtained values on the simulator will use the authors finding to determine if times and exposition on simulator where safe or not. Successful evacuations would be the ones done in times below the optimal time found for the route and the same rule applies for hazard exposition that should less than the ones obtained by the algorithm.

Table 4: Classification of all sessions results. Source: The research

CLASSIF.	TRAINING	COMPLEXITY	VISIBILITY	ROUTE TIME	HAZARD EXPOSURE
SUCCEED	HIGH	HIGH	NIGHT	ABOVE	ABOVE
FAIL	LOW	LOW	DAY	BELOW	BELOW

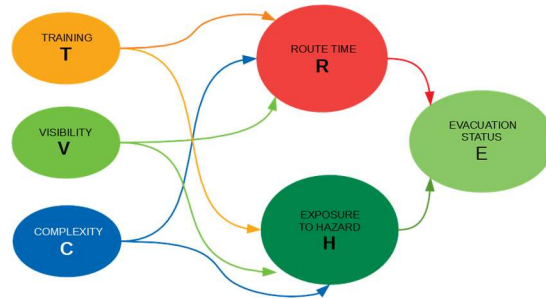
Finally, all the session data was put in a table following the formatting of Table 4 with all results following a classification status, permitting to verify the frequency of every one of the independent variables. Each line reported all context conditions in each one of the simulator sessions. Then the names for each node on the network were defined, and in this case, the first letter of each variable in Table 5.

Table 5: Variables Denomination for Bayesian Nodes. Source: The Research

Variable	Node Denomination
Evacuation status	E
Training	T
Complexity	C
Visibility	V
Route Time	R
Hazard Exposure	H

The next steps would consist in feeding the Bayesian Network (Figure 9) with the obtained data and generate the conditional probabilities tables (CPT) for the scenario a process very well covered in [22]–[24]. This step was left for future works.

Figure 9: Directed Acyclic Graph (DAG) with dependencies between nodes. Source: The Research.



The created simulator worked properly, with the analytics events written to disk in a JSON file in real-time as the player passed through the smart sensor, giving all information about the conditions on the route. The selected events also triggered google analytics events permitting telemetry over the web. The developer has the freedom to record all data with no processing at all or do, inside the GE, some pre-processing to the data before recording it to the log files. Either way, was tested and worked flawlessly.

For outputted for data consistency, special attention to units must be payed and is recommended to run a few sessions and test the numbers obtained, before committing the definitive and more extensive sessions. In the test sessions even though the sample sizes were not the ideal ones, identifying and addressing issues will save valuable time in the experiment.

After the conversions step, the outputted data becomes an MS Excel file permitting further processing, testing, and analysis.

4. CONCLUSIONS

The overall process was very successful, using GEs to build VE is a quite straightforward process, all the graphics display parts and a great amount of physics simulation can be handled inside the GE. One main issue to be kept in mind is that a GE is not a dynamic simulator, it don't has the precision in the results for simulating complex dynamic processes. This limitation can be compensated by using data interchange formats as was used in the present work to input the concentrations simulated in Aloha[™]. Respected this limitation, the possibilities are endless, and with the bonus of always starting with a good-looking environment.

References

- [1] M. Musharraf, D. Bradbury-Squires, F. Khan, B. Veitch, S. MacKinnon, and S. Imtiaz, "A virtual

- experimental technique for data collection for a Bayesian network approach to human reliability analysis,” *Reliab. Eng. Syst. Saf.*, vol. 132, pp. 1–8, Dec. 2014.
- [2] Y. Ba, W. Zhang, G. Salvendy, A. S. K. Cheng, and P. Ventsislavova, “Assessments of risky driving: A Go/No-Go simulator driving task to evaluate risky decision-making and associated behavioral patterns,” *Appl. Ergon.*, vol. 52, pp. 265–274, Jan. 2016.
 - [3] D. Cohen *et al.*, “Emergency preparedness in the 21st century: Training and preparation modules in virtual environments,” *Resuscitation*, vol. 84, no. 1, pp. 78–84, 2013.
 - [4] D. Cohen *et al.*, “Tactical and operational response to major incidents: Feasibility and reliability of skills assessment using novel virtual environments,” *Resuscitation*, vol. 84, no. 7, pp. 992–998, 2013.
 - [5] Visiongain, “The Oil & Gas Virtual Reality (VR) Training and Simulation Market 2011-2021 The Oil & Gas Virtual Reality (VR) Training and Simulation Market 2011-2021,” 2011.
 - [6] Immerse inc, “Immerse Learning - UK,” 2017. .
 - [7] L. Bishop, D. Eberly, T. Whitted, M. Finch, and M. Shantz, “Designing a PC game engine,” *IEEE Comput. Graph. Appl.*, vol. 18, no. 1, pp. 46–53, 1998.
 - [8] R. Darken, P. McDowell, and E. Johnson, “Projects in VR: The Delta3D open source game engine,” *IEEE Comput. Graph. Appl.*, vol. 25, no. 3, pp. 10–12, 2005.
 - [9] Crytek GmbH, “Cry Engine,” 2017. [Online]. Available: <https://www.cryengine.com>.
 - [10] I. Epic Games, “Unreal Engine,” *Unrealengine.Com*, 2017. [Online]. Available: <https://www.unrealengine.com/what-is-unreal-engine-4>.
 - [11] M. Cha, S. Han, J. Lee, and B. Choi, “A virtual reality based fire training simulator integrated with fire dynamics data,” *Fire Saf. J.*, vol. 50, pp. 12–24, 2012.
 - [12] V. Nyamse, V. Charissis, J. D. Moore, C. Parker, and S. Khan, “The Design Considerations of a Virtual Reality Application for Heart Anatomy and Pathology Education,” in *15th International Conference, VAMR 2013 Held as Part of HCI International 2013 Las Vegas, NV, USA, July 2013, Proceedings, Part II*, 2013, vol. 8021, pp. 66–73.
 - [13] L. Chittaro and R. Sioni, “Serious games for emergency preparedness: Evaluation of an interactive vs. a non-interactive simulation of a terror attack,” *Comput. Human Behav.*, vol. 50, pp. 508–519, 2015.
 - [14] E. Adams, *Fundamentals of Game Design*, 2nd ed. Berkeley, CA: New Riders, 2013.
 - [15] S. Brambilla and D. Manca, “Recommended features of an industrial accident simulator for the training of operators,” *J. Loss Prev. Process Ind.*, vol. 24, no. 4, pp. 344–355, 2011.
 - [16] D. Manca, S. Brambilla, and S. Colombo, “Bridging between Virtual Reality and accident simulation for training of process-industry operators,” *Adv. Eng. Softw.*, vol. 55, pp. 1–9, 2013.
 - [17] N. J. Ekanem, A. Mosleh, and S.-H. Shen, “Phoenix—A model-based Human reliability analysis methodology: Qualitative analysis procedure,” *Reliab. Eng. Syst. Saf.*, vol. 145, pp. 301–315, 2015.
 - [18] E. Oliveira da Silva, “DEVELOPMENT OF A MULTI-OBJECTIVE GENETIC ALGORITHM TO REDUCE INDIVIDUAL RISK AND DEVELOPMENT OF A MULTI-OBJECTIVE GENETIC,” 2017.
 - [19] E. Alp, M. J. E. Davies, R. G. Huget, L. H. Lam, and M. J. Zelensky, “GASCON2 - A Model to Estimate Ground-level H2S and SO2 Concentrations and Consequences from Uncontrolled Sour Gas Releases,” 1990.
 - [20] W. W. Perry and W. P. Articola, “Study to Modify the Vulnerability model of the Risk Management System - #,” Washington, D.C., 1980.
 - [21] W. mei Gai, Y. feng Deng, Z. an Jiang, J. Li, and Y. Du, “Multi-objective evacuation routing optimization for toxic cloud releases,” *Reliab. Eng. Syst. Saf.*, vol. 159, no. October 2016, pp. 58–68, 2017.
 - [22] J. B. D. Marco Scutari, *Bayesian Networks - With Examples in R*. 2015.
 - [23] J. Williamson and B. Nets, “Bayesian Nets and Causality,” *Book*, 2005.
 - [24] K. K. AGGARWAL, *Bayesian Methods in Reliability*, vol. 15. Dordrecht: Springer Netherlands, 2010.