Participatory and Evolutionary Fire Simulation via a Sensitive Control of Key Scenery Parameters

Qi Zhu[1]; Jing Du, Ph.D., A.M.ASCE[2]; Yangming Shi[3]; Qi Wang, Ph.D.[4]; and Yingzi Lin, Ph.D.[5]

[1]Ph.D. Student, Engineering School of Sustainable Infrastructure and Environment, Univ. of Florida, 454A Weil Hall, Univ. of Florida, Gainesville, FL 32611. E-mail: qizhu@ufl.edu
[2]Associate Professor, Engineering School of Sustainable Infrastructure and Environment, Univ. of Florida, 460F Weil Hall, Univ. of Florida, Gainesville, FL 32611 (corresponding author). E-mail: eric.du@essie.ufl.edu
[3]Ph.D. Student, Engineering School of Sustainable Infrastructure and Environment, Univ. of Florida, 454A Weil Hall, Univ. of Florida, Gainesville, FL 32611. E-mail: shiyangming@ufl.edu
[4]Assistant Professor, Dept. of Civil and Environmental Engineering, Northeastern Univ., Boston, MA 02115. E-mail: q.wang@northeastern.edu
[5]Associate Professor, Dept. of Mechanical and Industrial Engineering, Northeastern Univ., Boston, MA 02115. E-mail: yi.lin@northeastern.edu

ABSTRACT

Fire simulator training is a very effective way to reduce firefighter injuries and improve their mission performance. However, the usability of existing fire simulators remains an issue as most of these simulators require users to set up hundreds or even thousands of parameters to initiate the simulation. In this paper, we present a participatory and evolutionary fire simulation method that balances the accuracy and fidelity of the fire simulation and the usability. The base parameters for accurate and usable fire simulation can be selected by a combination of the evolutionary algorithm and the knowledge of domain experts via the virtual reality (VR) fire simulator. These base parameters are expected to expedite the use of fire simulators and reduce the adoption threshold.

INTRODUCTION

Fire-related accidents cause significant injuries, casualties and property loss in the US. According to the Federal Emergency Management Agency (FEMA), in 2017 there were more than 1,319,500 fire-related accidents in the US, causing 3,400 deaths, 14,670 injuries, and $23 billion property loss (FEMA 2017). Meanwhile, a total of 68,085 firefighter injuries occurred in the line of duty in 2015, and 42.8% of all firefighter injuries occurred during fire ground operations (Haynes and Molis 2015). One of the most effective ways of reducing firefighter injuries and improving their mission performance is through the use of fire simulator training (Cha et al. 2012). Fire simulators can also be used to plan out the response strategies in practice (Finney 1998). As a result, the use of fire simulators has drawn a great interest in both the academia and practices. For example, the Fire Dynamic Simulator (FDS) developed by National Institute of Standards and Technology (NIST) (McGrattan et al. 2010) has been widely used in evacuation planning (Korhonen et al. 2009), building designs (Hu et al. 2007), hazard assessment (Xu et al. 2014), and firefighter training (Cha et al. 2012).

Yet the adoption of the existing fire simulators is lagging due to multiple difficulties. A major difficulty is that many fire simulators, although excellent in fidelity and accuracy of their simulation, are overly complex and difficult to be implemented in practice. End users (e.g., firefighters or emergency response planners) often need to set values of hundreds, or sometimes, even thousands of parameters to initiate the simulation (McGrattan et al. 2013). For example, the users of the FDS have to set parameters about the combustion materials, room configurations, temperature and humidity and other fire and environmental factors to conduct the simulation (McGrattan et al. 2010), as these fire simulators rely on a physics engine to simulate the fire behaviors and development. The minimum inputs needed to simulate the physical processes are beyond many laypeople's capability. In this situation, simulation fidelity and accuracy are often at the cost of the usability.

Owing to the urgency and criticality of fire emergencies, there is a pressing need to find a balance between the accuracy of the fire simulation and the usability. The biggest challenge is to identify a small subset of fire simulation parameters that can still yield comparable and similar enough simulation effects as using all parameters needed by the popular physics engines. We call these parameters the base parameters for they represent the minimum number of key parameters for a satisfactory fire simulation without losing the flexibility of simulating various scenarios. When the number of base parameters are below a practical threshold, the increased usability is expected to improve the adoption of fire simulators among the front line users. To achieve this goal, this study proposes a participatory and evolutionary process to identify the base parameters for accurate and usable fire simulation. The knowledge of domain experts is solicited by tracking their setup preferences and behaviors while using an interactive Virtual Reality (VR) fire simulator. This interactive VR system was developed based on our previous systems (Du et al. 2016; Du et al. 2017; Du et al. 2017; Du et al. 2018; Shi et al. 2016). An evolutionary algorithm is then applied to assign weights to the simulation parameters based on an analysis of domain experts' behaviors. In the remainder of this paper the background and technical details of the proposed method will be introduced with a case demonstration.

LITERATURE REVIEW

Many fire simulators rely on a physics engine that simulates the physical behaviors of a fire event and the characteristics of its development (Cha et al. 2012). One of the most famous simulators in this category is Fire Dynamic Simulator (FDS) (Shen et al. 2008). FDS is a large-eddy simulation for low-speed flows, with an emphasis on smoke and heat transport from fires developed by NIST (McGrattan et al. 2010). On the basis of hydrodynamics, heat and mass transfer, combustion theory and other theories, an FDS-based simulator could predict fire simulation reliably. The fire simulators based on physics engines represent a high fidelity and high accuracy reproduction of real world fire development because it builds on the physics that rules the real world fire events. As a result, they have been widely used to predict the impact of different fire types on temperature, smoke layer and thickness, and heat release rate in the real-world (McGrattan et al. 2010). Yet this type of fire simulation requires the input of a mass of parameters and considerable computation time. Take the FDS for example, the input variables required to yield a high quality simulation are more than hundreds (McGrattan et al. 2010), making it impractical in real-world applications.

To overcome the use difficulty of physics engine based fire simulators, interactive simulation technology has gained its popularity to enable users to dynamically adjust the fire simulation using a small set of key parameters. The most representative of this technology is the particle system which expresses fluid flows into movements of fine particles (Hockney and Eastwood 1988). The basic idea of the particle system is that a mass of fine particles with certain life cycles and attributes are used as fundamental elements to simulate irregular fuzzy objects. In this system, every particle has an attribute which are shape, size, color, transparency, the velocity of movement and direction, life cycle and so on. Every particle is not a static whole, and it goes through three distinct phases as the generation, dynamics and death over time. The particle system based fire simulation has the following characteristics: flexible particle groups contain both simplest points and a bit of structure. So they can be adjusted freely by the size and the shape of the simulated object. It is great for showing dynamic features of realistic flame with random function as all its attributes vary over time. Besides, the hierarchical particle system can simplify the generation of various flames for its interactivity and controllability.

The two fire simulation methods indeed represent two simulation paradigms in general, i.e., the bottom-up paradigm and top-down paradigm. The bottom-up simulation paradigm models the fundamental elements of a system and then allows system behaviors to emerge during the simulation (Bonabeau 2002). The physics engine based fire simulators rely on the fundamental physics rules of the fire; before the simulation is finished, the modeler does not have any knowledge about how the fire would evolve, neither the knowledge about what fire parameters are critical. In contrast, the interactive fire simulation represents the top-down simulation paradigm that focuses on identifying a set of predefined parameters to mimic the system behaviors to the maximum extent (Bukowski and Sequin 1997). Both paradigms have their own pros and cons in fire simulation: bottom-up simulation usually yields more accurate results while top-down simulation is more practical given its better usability. This study proposes a method to strike a balance between the simulation accuracy and usability with a participatory process. Details will be discussed in the following section.

PROPOSED PARTICIPATORY AND EVOLUTIONARY FIRE SIMULATION

We propose a participatory process to identify the most critical fire simulation parameters which, once been adjusted properly, will yield a fire simulation comparable to the physics engine based fire simulation. The technical challenge is to identify the small-enough subset of fire simulation parameters to not only enable accurate representations of the possible fire scenes but also provide an improvement in flexibility of the representation, i.e., the minimum number of parameters. To achieve this goal, we present an evolutionary and participatory method described as below.

Step 1: Develop a VR based fire simulation platform. VR provides the immersive experience that triggers realistic psychological responses of the users (Shi et al. 2018). Besides, a real-time visual representation of fire event in VR based fire simulation platform combining with 2D control panel with multiple parameters allows a fire expert to have a better understanding of the connections between fire event and parameters. We the fire expert as people who has more than 20 years working as a firefighter. They should be qualified to evaluate the fire and life safety and have the expertise in training of fire professionals in order to testify whether the simulation results from the system is appropriate for fire fighter training. In the way, the experts can use various 3D fire visualizations to get an insight into the simulation results and relate those results to the fire event in the real world.

Step 2: Categorize fire events. A fire event could happen in different forms and develop in different ways depending on the cause and the environment (Havel 2013). According to the National Fire Protection Agency (NFPA), a fire event can be affected by the types of fuels, the development stage, and the spread in the fire-fighting field. In this paper, we translate them into a three-dimensional categorization system including *Static attributes*, *Dynamic development features*, and *Building types* to differentiate various flame scenes in the real world. Before using our system to simulate a fire, the user needs to select a specific fire event from this system as the target scene.

Step 3: Develop an initial list of control parameters. To facilitate the benchmarking of simulation performance parameters, the following requirements must be satisfied: (1) the parameters must be related to categorized fire events; and (2) the parameters can be visualized with the particle system in the game engine. We propose to integrate the following parameters (see Table 1) in the VR simulation platform.

Table 1. Example control parameters used in the analysis

Category	Explanation	Parameters	Description
Static	Related to the static attributes of the fire event	Max particles	The maximum number of particles the system may have alive at once.
		Start rotation	The initial rotation angle of the particles.
		Start color	The initial color of the particles.
		Shape	The shape of the emission volume.
		Angle	The angle of the cone at its point. An angle of 0 produces a cylinder while an angle of 90 gives a flat disc.
		Radius	The radius of the circular aspect of the shape.
Dynamic	Related to the development of the fire event	Duration	The length of time the system runs.
		Start delay	The delay in seconds before the particle system starts emitting.
		Start lifetime	The initial lifetime in seconds for the particles. The particle is destroyed after this elapsed time.
		Start speed	The initial speed of the particles. The greater the speed of the particles, the more spread out they will be.
		Looping	If enabled, the system starts again at the end of its duration time and continues to repeat the cycle.
		Rate over time	The number of particles emitted per unit of distance moved.
Building	Related to the environment of the fire event	Fire-resistive	Permitted combustibles as trim and furnishings.
		Non-Combustible	Combustible trim and furnishings; may have combustible non-loadbearing partition walls.
		Ordinary	Combustible interior partitions and load-bearing walls, floors, roofs, trim, and furnishings, with exterior walls of masonry
		Heavy timber	Combustible (but not easily ignitable) columns, beams, floors, roofs, trim, and furnishings; the load-bearing and exterior walls are masonry

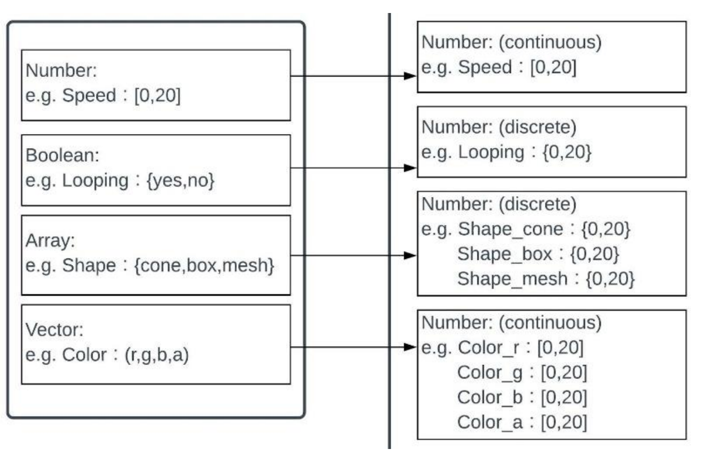

Figure 1. A representation of the mapping layer (arrows) between original parameter data and the output variables after converted

In order to efficiently analyze the relationship between input and output variables, we convert all data types into the numeric type of the same range. This includes creating tables and establishing the relationships among those tables according to the rules listed in Figure 1, which are designed both to protect the data and to make the database more flexible. For example, if a parameter 'Looping' with Boolean value as 'True' and 'False', we will assign number type values to the original ones like 0 to 'False',20 to 'True'. After transformation, the value set of 'Looping' contains only two standards numeric values (0 and 20). Similarly, if the parameters with array values, we transfer this type of values from two steps. Firstly, we separate every component of the original set. Then these values will be converted into Boolean values using the way mentioned above.

Step 4: Seek expert inputs in a "*what you see is what you get*" manner, i.e., adjusting parameters as simulation proceeds. In this paper, the simulation begins with a specific fire event in a certain building environment. In step 2, we have proposed how such a fire event can be defined, based on its physical and environmental features. In this step, the domain expert is asked to set the value of given parameters to make sure that the 3D fire graph represents a real fire event as closely as possible. In the case of multiple simulation runs, the expert starts to simulate the fire event from the first stage of fire (Ignition) to the fourth stage (Decay). On each stage, the expert can adjust the simulation with various combinations of the control parameters until getting the final result.

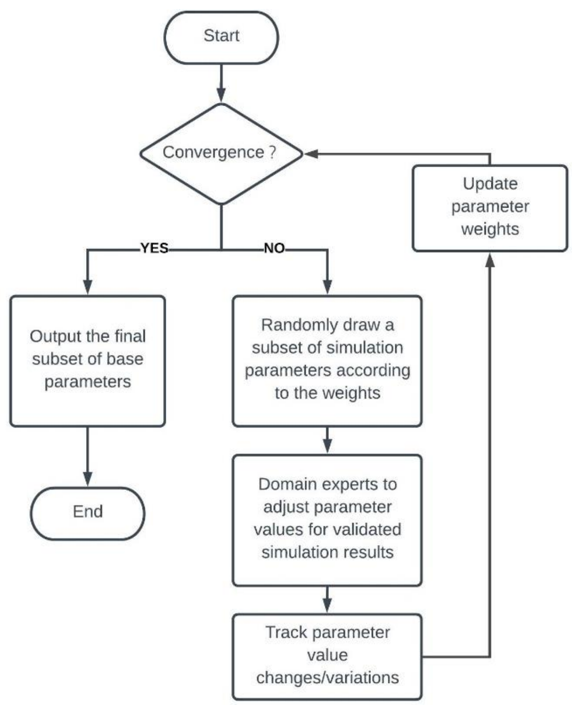

Figure 2. Flow of the evolutionary and participatory fire simulation method

Step 5: Sensitivity analysis to rank the relative importance of the selected parameters. We propose a generalized framework (Figure 2) for ranking the selected parameters. Our framework employs an iterative approach to find a subset including parameters most frequently used. We assume that X is the set of parameters defined for the simulator interface and C_k is a subset of X in *k-th* literation: $C_k = \{x \mid x \subset X\}$. We initialize the interface by randomly generating m parameters in the dataset X. Each iteration of our analysis approach consists of two steps:(1)

track selected parameter value changes, update weights and assign them to selected parameters (2) weighted randomly select parameters. The function of changes in every selected parameters in *k-th* literation is defined as $\Delta x_i = \| x_i^k - x_i^o \|^2$, where x_i^o denotes the original value of parameter x_i, and x_i^k denotes the value after adjusting by experts in *k-th* literation. The goal is to find the x^* that maximizes its changing values in every iteration, where $x^* = \arg\max_k \{\Delta x\}$. Then, we weight x^* and update set X with weight value. Now we get a set X of n weighted items, and then we randomly select samples, which the probability of each item to be selected is determined by its relative weight. We use the algorithm based on WRS(Weighted random sampling) (Efraimidis and Spirakis 2006) to get our weighted samples. By this algorithm, we calculate the score of x_i by the function as $S_i = R^{\frac{1}{w_i}}$, where w_i denotes the weight of item x_i, and R denotes a random number generated from $random(0,1)$. Then we select the m largest items with the largest score S_i as a set with a weighted random sample (Efraimidis and Spirakis 2006). The weights update and weighted randomly select steps are carried out iteratively until we reach a fixed set. Practically speaking, we assume that this approach has converged if $\max_{i=1}^{n} \{w_i\} \geq \epsilon$, where n is the total of all parameters and ϵ is the convergence threshold. With convergence to an ideal state, we can expect that the parameters in the fixed set are base parameters we want to figure out.

SYSTEM DEMONSTRATION

The system will provide experts with a user-friendly simulation as well as functions to collect and analyze data at the same time. It contains three components: Simulation View (SV), Interaction View (IV) and a Cloud server that transfers data and facilitates the parameter optimization algorithm.

Simulation View (SV): The SV (Fig.3a) consists of two parts: (1) Building environment (BE) including walls, floors and other essential constituents of architectural place from the aspect of the definitions of space, field, and place. (2) Fire and smoke simulation (FSS) based on the particle system. We use building environment to simulate the scene of the fire. As the particle system is designed to collide with the architectural elements in SV such as walls, so we can employ a visual display simulating how these building constituents interact with fire events. Besides, BE and FSS is designed to be two separate components, so we can completely replace current BE with other new scenes to simulate different fire events.

a. Simulation View b. Interaction View

Figure 3. An example of the developed system view: fire scenario in the subway

Interaction View (IV): The IV (Fig.3b), is an interactive interface system that is designed to help experts control fire event by adjusting scrollbars of different parameters. The resulting rendered user interface is depicted on the right side of Fig. 3b. The expert can get immediate feedbacks of visual changes of fire event once they have dragged the scrollbars. A back-end analytics system applied the algorithm introduced in the previous section helps track and calculate the date.

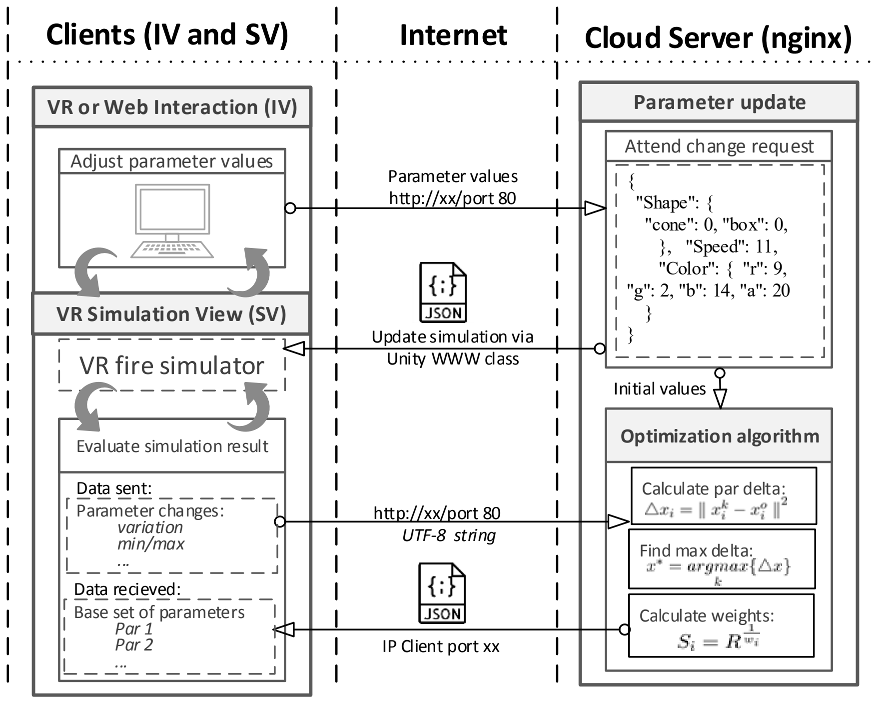

Figure 4. System architecture of the participatory and evolutionary fire simulation

Cloud Server: The Cloud server via *nginx* apps is used to transfer metadata and perform the parameter optimization algorithm to find out the base parameters (Fig.4). Jason is used as the data transfer medium. First, a VR or Web Brower based IV is used by the domain experts to initiate parameter values. The values are sent to the server as data.json via HTTP. Then the data.json will be read by Unity via the WWW method, i.e., WWW data=WWW (url). The SV is updated based on the json data. Domain experts will evaluate the SV and further adjust the parameter values. The updates will be sent to the server via data.json, which will be compared against the initial values. The weights of the parameters will be updated based on deltas. Ultimately the result (i.e., the base parameters) will be returned to the SV via Jason.

DISCUSSION AND CONCLUSIONS

The use of fire simulators has gained significant momentum in recent years. Especially in emergency response and planning, fire simulators provide an insight into the characteristics and the development of various fire events so that the firefighters have reliable information before entering into the site. Yet a further investigation finds that most fire simulators are limited in their applicability due to their complex setup. A popular approach in fire simulation is to build around a physics engine that reproduces the complex physical processes of fire and smoke development. To obtain a good simulation result, users usually need to set values of hundreds of or even thousands of parameters. In this study, we propose a participatory and evolutionary simulation paradigm that relies on the holistic view of the domain experts on a fire event. Only a list of selected key scenery parameters, called *base parameters*, are used to generate a simulation comparable to the fire simulators based on physics engines. This proposed paradigm is expected to significantly expedite the use of fire simulators and reduce the adoption threshold. The validated key parameters are also expected to simplify the existing fire simulators relying on physics engines. Due to the length of the paper, there exist some limitation. The numerical results and validation methods will be discussed in the future research

ACKNOWLEDGEMENTS

This material is supported by National Science Foundation (NSF) under Grants 1761459 and 1761950, as well as National Institute of Standards and Technology (NIST) under Grant 60NANB18D152.

REFERENCES

Administration, U. S. F. (2017). "U.S. fire statistics." <https://www.usfa.fema.gov/data/statistics/>.

Bukowski, R., and Sequin, C. "Interactive simulation of fire in virtual building environments." *Proc., Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., 35-44.

Cha, M., Han, S., Lee, J., and Choi, B. J. F. S. J. (2012). "A virtual reality based fire training simulator integrated with fire dynamics data." 50, 12-24.

Du, J., Shi, Y., Mei, C., Quarles, J., and Yan, W. "Communication by Interaction: A Multiplayer VR Environment for Building Walkthroughs." *Proc., Construction Research Congress 2016*, 2281-2290.

Du, J., Shi, Y., Zou, Z., and Zhao, D. (2017). "CoVR: Cloud-Based Multiuser Virtual Reality Headset System for Project Communication of Remote Users." *Journal of Construction Engineering and Management*, 144(2), 04017109.

Du, J., Zou, Z., Shi, Y., and Zhao, D. (2017). "Simultaneous Data Exchange between BIM and VR for Collaborative Decision Making." *Computing in Civil Engineering 2017*, 1-8.

Du, J., Zou, Z., Shi, Y., and Zhao, D. (2018). "Zero latency: Real-time synchronization of BIM data in virtual reality for collaborative decision-making." *Automation in Construction*, 85, 51-64.

Efraimidis, P. S., and Spirakis, P. G. J. I. P. L. (2006). "Weighted random sampling with a reservoir." 97(5), 181-185.

Finney, M. A. J. R. P. R.-R.-., Revised . Ogden, UT: US Department of Agriculture, Forest Service, Rocky Mountain Research Station. 47 p. (1998). "FARSITE: Fire Area Simulator-model development and evaluation." 4.

Havel, G. (2013). "Construction Concerns: Construction Types and Fire Behavior."

Haynes, H. J., and Molis, J. L. (2015). *US firefighter injuries-2014*, National Fire Protection Association. Fire Analysis and Research Division.

Hockney, R. W., and Eastwood, J. W. (1988). *Computer simulation using particles*, crc Press.

Hu, L., Fong, N., Yang, L., Chow, W., Li, Y., and Huo, R. J. J. o. H. M. (2007). "Modeling fire-induced smoke spread and carbon monoxide transportation in a long channel: fire dynamics simulator comparisons with measured data." 140(1-2), 293-298.

Korhonen, T., Hostikka, S. J. T. R., and Finland, U. s. G. V. T. R. C. o. (2009). "Fire dynamics simulator with evacuation: FDS+ Evac."

McGrattan, K., Hostikka, S., Floyd, J., Baum, H., Rehm, R., Mell, W., and McDermott, R. J. N. s. p. (2010). "Fire dynamics simulator (version 5), technical reference guide." 1018(5).
McGrattan, K., Klein, B., Hostikka, S., and Floyd, J. J. N. s. p. (2010). "Fire dynamics simulator (version 5), user’s guide." 1019(5), 1-186.
McGrattan, K. B., McDermott, R. J., Weinschenk, C. G., and Forney, G. P. (2013). "Fire dynamics simulator, technical reference guide."
NFPA (2018). "Reporter's Guide: All about fire." URL: https://www.nfpa.org/News-and-Research/News-and-media/Press-Room/Reporters-Guide-to-Fire-and-NFPA/All-about-fire#class. Access Date: Oct 15, 2018.
Shen, T.-S., Huang, Y.-H., Chien, S.-W. J. B., and environment (2008). "Using fire dynamic simulation (FDS) to reconstruct an arson fire scene." 43(6), 1036-1045.
Shi, Y., Du, J., Ragan, E., Choi, K., and Ma, S. "Social Influence on Construction Safety Behaviors: A Multi-User Virtual Reality Experiment." *Proc., Construction Research Congress 2018*, 174-183.
Shi, Y., Du, J., Lavy, S., and Zhao, D. (2016). "A Multiuser Shared Virtual Environment for Facility Management." *Procedia Engineering*, 145, 120-127.
Xu, Z., Lu, X., Guan, H., Chen, C., and Ren, A. J. A. i. e. s. (2014). "A virtual reality based fire training simulator with smoke hazard assessment capacity." 68, 1-8.