Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

INCORPORATING SOUND IN SIMULATIONS

Justin Deuro

Modeling, Simulation, and Visualization Engineering Dept.
Old Dominion University
1300 Eng. & Comp. Sciences Bldg.
Norfolk, VA 23529, USA

Christopher J. Lynch

Virginia Modeling, Analysis, and Simulation Center
Old Dominion University
1030 University Boulevard
Suffolk, VA, USA

Hamdi Kavak

Modeling, Simulation, and Visualization Engineering Dept.
Old Dominion University
1300 Eng. & Comp. Sciences Bldg.
Norfolk, VA 23529, USA

Jose J. Padilla

Virginia Modeling, Analysis, and Simulation Center
Old Dominion University
1030 University Boulevard
Suffolk, VA, USA

ABSTRACT

In this paper, we raise the questions: how could sound influence the usability of simulations? How could sound influence the learning of simulation creation? How could sound support processes like verification? We argue that sound can support learning by relying on music/sound cues' emotional engagement on users and verification by providing insight into the correctness of simulation execution during runtime. For instance, sound cues could indicate when certain events occur and if processes in a simulation are operating within their specifications. We explore potential benefits and challenges posed by incorporating sound into DES models. Many perceived challenges of this incorporation overlap with known visualization challenges for conveying information during runtime, as both cases deal with conveying sensory stimuli. We present conceptual examples and report on ongoing efforts to integrate sound into a DES simulation environment.

1 INTRODUCTION

Sounds play a major role in experiencing, interacting, responding to, and receiving feedback from computers. Noises indicate incoming calls or texts, voices providing audible instructions while following navigation systems, and music helps to establish moods within games. The use of sounds helps to associate the occurrence of specific events with their intended audiences and teach these audiences to react in a certain way. For instance, the video games of the 1980s used specific types of sound cues to indicate different types of events. In Nintendo's *Mario Bros.*, special noises indicated when the character died or when successfully completing a level. A special noise indicates the start of a player's final lap in Nintendo's *Super Mario Kart* while alarms sound in Maxis's *SimCity 2000* when disasters start to occur within a city. A common audio component of many games is to increase the tempo of the game's background music when the player is almost out of time to complete the level.

Along with games, movies and other media also incorporate the use of sound cues to affect the moods and messages portrayed. The choice of music used has been shown to have great influence over mood responses (Stratton and Zalanowski 1991) and is actively used to influence marketing of consumers

(Bruner 1990). We believe that sound cues, within the context of simulation, can provide benefits such as notifying the modeler that the simulation is not adhering to its specifications, that system failures consistently occur at a specific location, or that everything occurs as expected. In this manner, sound cues can contribute to modelers' abilities to conduct verification by using audible confirmation to check that their simulations adhere to intended specifications.

Sound cues for conveying sensory indication of simulation progression can be implemented across modeling paradigms within Modeling and Simulation (M&S). For instance, within Discrete Event Simulations sound allows for tracking if queue sizes become abnormally large based on the model's specifications. In System Dynamics, sound can alert the user when stock values move too far from the expected value. Within Agent Based Models, sound can indicate when certain interactions occur; for example, in the predator-prey model, a death sound can indicate when a predator eliminates a prey agent.

Despite the potential benefits, sound cues are not commonplace components of simulation models. Simulation environments rarely provide opportunities to incorporate sound into the execution of the simulation. To the best of our knowledge, no simulation engine currently provides this functionality as a default feature. Finally, there is no systematic study investigating the use and effect of sound cues in building or using simulation models.

In this paper, we explore the question *how could sound influence the usability of simulations*? We argue that the incorporation of sound has potential positive implications on simulation learning and in processes like verification. We report on ongoing sound capability development on the CLOUDES simulation environment (Padilla et al. 2014). We expect the integration of sound and simulation to add a sensory dimension to the creation and execution of simulations and to extend the user base of M&S to underserved communities like the visually impaired.

2 BACKGROUND

Simulations provide an efficient and effective way to teach in a more hands-on approach to learning than reading a textbook or listening to a lecture. Simulation engines are becoming easier to access and more pervasive through the availability of online tools, the ability to collaborate on projects online, and to easily share models online (Fortmann-Roe 2014; Padilla et al. 2014). To further enhance the process of teaching and learning with simulation, games provide a technique for engaging learners in the educational process. Games can serve as an effective instrument in teaching simulation topics in an engaging manner for the learner as multiple disciplines (Moore, Chamberlain, Parson, and Perkins 2014; Padilla et al. 2016). Incorporating sounds into game-based simulation can enhance the sensory stimuli of the game and further connect the simulation user to the real system and help to illuminate critical topics within training sessions, reinforce complicated course material, and create associations to real sounds within the minds of the learners. Additionally, sound cue research provides a step towards the creation of methods for assisting visually impaired individuals in building and executing simulations.

Simulation environments rarely provide opportunities to incorporate sound into the execution of the simulation. Sound cues could provide benefits such as notifying the modeler that the simulation is not adhering to its specifications, that system failures consistently occur at a specific location, or everything occurs as expected. In this manner, sound cues can contribute to modelers' abilities to conduct verification by using audible confirmation to check that their simulations adhere to intended specifications. Since sound cues are sensory inputs, we believe that challenges in implementing these features and the benefits that sound cues can add to the verification process mirror the benefits and challenges presented by the use of visualizations for these same purposes, including usability, scalability, and misrepresentation. These items drive our discussion in Section 4. To the best of our knowledge no simulation engine currently provides this functionality as a default feature.

Sound is considered an "all around" sense instead of a directional sense like visual stimuli as explained in (Rauterberg and Styger 2005). In using sound to explain ideas, a much more general feeling of what is happening around you (i.e. in the simulation) can be achieved. With visual cues, only the information that you are currently paying attention to is related and key material can be conveyed back to the users while filtering out unnecessary information on the screen. Having sound present in a simulation allows for information to be given from outside context.

Several studies have used music to assist in teaching specific topics. Nursing teachers create catchy songs to train healthcare students and professionals with the lyrics conveying important components of the course material, such as the proper sequences for assessments and interventions (Bowen 1999). Teachers in Economics uses song lyrics as an effective aid to teaching and engaging introductory level concepts at the undergraduate level (Tinari and Khandke 2000). Similarly, Hall and Lawson (2008) use song lyrics to engage students in the concepts of Microeconomics and found that this helped to engage students even though an increase in learning was not observed in all students. Levy and Bryd (2011) find that music has the potential for teaching concepts in Social Justice.

From the sensory perspective, information provided to model users during runtime commonly include visual components. Sound cues can enhance the representations of the model execution that are conveyed to its users similar to how visualizations are used to accomplish this same goal. Animation provides the modeler the ability to see how the modeled system functions and helps the modeler to understand the system's behavior and helps the modeler to communicate model results to others (Rohrer 2000). Additionally, animation allows for the graphical representation of internal and external behaviors and helps modelers visually identify errors in the implementation (Balci 1998; Xiang et al. 2005). Operational graphics provide visual insights into dynamic model behaviors such as queue sizes and the percentage of busy resources over time (Sargent 2013). Visualization contributes to conceptualization, quantitative, exploration, and pattern and flow analysis within simulation studies (Vernon-Bido, Collins, and Sokolowski 2015). All of these characteristics also apply to the use of sound within simulations.

3 INCORPORATING SOUND CUES INTO A DES MODEL

Sound cues can enhance the representations of the model execution that are conveyed to its users similar to how visualizations are used to accomplish the same goal. Wenzel, Bernhard, and Jessen (2003) identify several visualization requirements to enhance the execution of a warehouse inventory model including representing stocks on fixed time intervals, representing stock levels whenever a change occurs, representing the distribution of stocks, visualizing the inbound and outbound amounts for the warehouse as a network relationship, and visualizing the interdependencies of the warehouse. Sound cues can replace or be used in conjunction with these types of visual aids to further enhance the user experience while using simulation models. For instance, representing the value of a variable over time could be handled by a periodic note sounding and the tempo or pitch of the note could alter based on changes to this value over time. Logical conditions could be used to set off a sound when specific events occur within the simulation such as a variable changing value, a variable exceeding a threshold, or a queue becoming empty.

DES model structure is simple which facilitates implementing sound into parts of a model. The addition of sound cues can improve DES model implementations by providing a more intuitive understanding of the behavior of the system during runtime. We implement sound in the CLOUDES environment through the initial use of third-party application programming interfaces (APIs). The examples of the different areas to include sound cues are proposed as additions to CLOUDES' current capabilities, and are intended as potential add-ons for any models.

Sounds are initially implemented at the module level. The user implements the triggering mechanism under different conditions per module. For example, when a queue reaches the specified number given by a modeler, the sound can begin to activate. Individual modules of the same type, such as process modules, can be assigned individual sounds. Otherwise, all modules of the same type share the same sound. Sounds are activated based on acceptable range values as highlighted in the following section.

3.1 Acceptable Range

To provide proper feedback for the nature of each individual module, when modelers fill in the details of each module, they will have the option to create an *Acceptable Range* value for that module. This is easiest to explain through the queue module. Already, the queue module contains the parameter *Max Size* which provides an option to limit the amount of entities within a specific queue. We suggest adding an *Acceptable Range* parameter to the queue module's options which requires a range of numbers that CLOUDES uses to determine if the queue moves outside of its specified range and the magnitude of this movement throughout execution. Having this as a range in all the pertinent modules enables the sounds to be played intermittently while within the acceptable range at constant time intervals, and then begin to play differently as the modulations occur, discussed below. Figure 1 provides a potential design for incorporating the *Acceptable Range* into the current CLOUDES display.

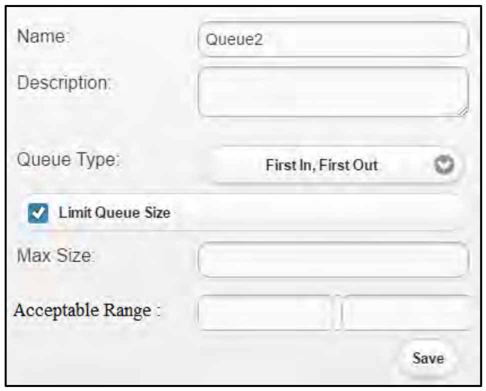

Figure 1: Example of the CLOUDES' Queue module updated to include the *Acceptable Range* input parameter (text boxes are added for illustrative purposes).

For process modules, the unit of measurement for *Acceptable Range* can refer to the percent utilization of resources and this simultaneously negates a need for an *Acceptable Range* parameter within the Resource module. The Batch and Separate modules can operate on a toggle, activating their sound when a batch or separate operation completes. Decisions can also operate using a toggle rather than a range with the sound option dependent upon the decision selected. For the *By Chance* option, the sound can occur whenever the least-likely or most-likely options occur.

3.2 Sound Banks

We propose the addition of three possible types of sounds: natural, synthetic, and musical. Natural sounds are considered to be sounds that occur in the real world, without computer manipulation, that can be recorded live. The natural sounds are chosen to both correlate to the type of module they're associated with, along with the ease of use of their sound. It is important to keep the frequency of all noises over a wide range to increase audio fidelity and prevent differing sounds from becoming muddled, a point brought up in Gaver et al. (1991).

Musical sounds are musical instrument implementations. There are several options in this variation. The first being creating chords with one instrument. In Table 1, a single instrument, such as piano, would be assigned a specific note of a specific chord. This way, the chord playing would signify normal operation. Modulated notes, according to the values falling out of the acceptable range, can reveal potential errors. Another implementation can assign specific arpeggios to modules that play at a set scale and speed during normal operation. Upon approaching the acceptable range values, the arpeggios can change in several ways. We discuss these changes the next section.

Table 1: Sound representations for the modules while operating within their specified Acceptable Ranges.

CLOUDES Module	**Natural**	**Synthetic**	**Musical**
Queue	Light Voices	Bell tone	Root Note
Process	"Now Serving"	Charge	Second
Resource	Heartbeat	Zap	Third
Batch	Water drop	Two-tone	Fourth
Separate	Bubble pop	Discharge	Fifth
Decision	Service bell	Ping	Sixth

Synthetic sounds are a middle ground between the natural sounds and the musical sounds. Natural sounds are easy to identify from their real world counterparts, and musical notes should be intuitive to most listeners. Synthetic sounds describe machine-made noises; sound effects used in movies and other types of simulations. These are readily available in several types of sound repositories and can be recreated easily. Table 1 describes how sounds can apply to each of CLOUDES' blocks. Figure 2 provides a potential view of a CLOUDES simulation *Settings* menu updated to include options for the user to select sound type.

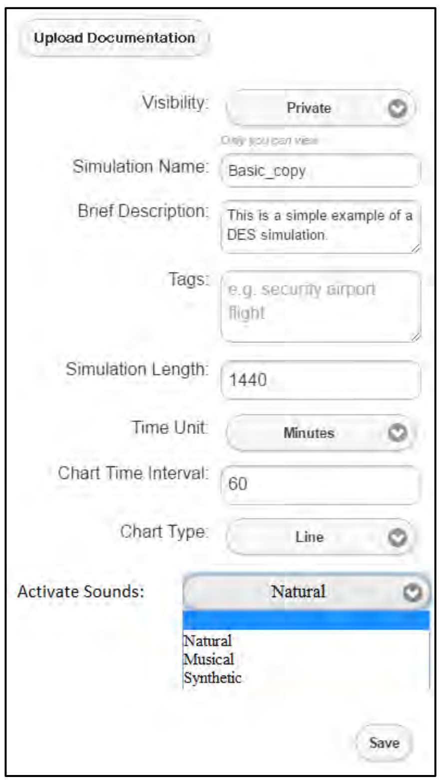

Figure 2: CLOUDES *Settings* menu updated to include additional sound type selection drop-down menu (dropdown menu is added for illustrative purposes).

3.3 Sound Modulation

When the parameters of any individual module fall outside of their *Acceptable Range* specifications, then the sound can begin to modulate. Since the *Acceptable Range* values are represented as discrete values, we can change the sounds based on the percentage that they differ from the acceptable values. As the values pass and begin to fall more out of line with the accepted range, the sounds will continue changing in order to draw more attention to the simulation module where the problem originates.

Table 2 shows examples of how a default sound can be replaced once the module is operating 10-20% outside of its expected range. For queues, increased numbers of waiting entities can be indicated by an increase in sound volume along with increasing the number of voices within the sound. Processes can use spoken warnings to indicate too high processing times or resource utilizations. A constant heartbeat can play as an underlying sound for resources with speed indicating utilization. Batches can trigger sounds exclusively when forming a grouped entity. Decision modules can use different sounds to indicate which paths are being taken by the entities.

Table 2: Sound representations when the modules operate 10-20% outside of their *Acceptable Range*.

MODULE	NATURAL
Queue	Medium voices
Process	"Process slowed" (spoken phrase)
Resource	Heartbeat 15% faster
Batch	-
Separate	-
Decision	Service bell

When module values begin to reach critical overages of greater than 20%, the natural sounds will incur large changes to show how out of step the modules have become, as shown in Table 3.

Table 3: Sound representations when the module operations exceed 20% of their *Acceptable Range*.

MODULE	NATURAL
Queue	Large Crowd
Process	"Process halted" (spoken phrase)
Resource	Rapid Heartbeat

In the case of the Synthetic and Musical sound banks, the modulations become much more elegant. For synthetic sounds, speeds and pitches are easily changed at will; tuning the pitches up around 20% and 50% for the 10-20% and >20% ranges respectively creates drastically different sounds that stand out from their normal operating counterparts.

With the musical sound bank, there is a great amount of flexibility in the manipulation of the instruments. For setting up a single chord for each module, individual notes can be played out of tune using pitch bend techniques. The bent pitch can vary several semitones between the original notes and can be discretely defined for use with the acceptable range critical values. Having a single note within a chord out of tune creates a very noticeable disharmony in the sound. This draws attention to the module assigned to this chord.

Using the arpeggio implementation, having any amount of speed changes becomes immediately recognizable by the simulator. For the lower critical value range, a tempo change of 15% causes enough of a disturbance to be irritating, and once the 20% acceptable range figure is passed, having a tempo increase of 35% creates a sense of urgency for that specific module, drawing great attention to it. In addition to the tempo changes, arpeggios can also have single notes shift out of tune to further increase the scope of the change for the acceptable range.

3.4 Module Assignments

Choosing a sound bank for the CLOUDES model within the *Settings* menu activates the sound cues within all of the simulation's modules. Additionally, there can also be an option to alter the specific sounds attached to each module. This is most apparent for a decision module, which can have different assigned sounds for each decision path to be made. A challenge occurs in the model user being able to differentiate between each module and easily identifying which one is falling outside of its range when using numerous sounds simultaneously or when assigned the same sound to multiple modules. For instance, if there are three different queues, and the queue sound becomes modulated, there must be a way to easily determine which queue is being effected. This can be achieved by sequencing sounds to play in the same order in which they were named within the simulation. Sound intensity can be chosen depending on the importance of individual modules. If sounds are chosen on the module level (where a single sound applies to all modules of that type rather than individual modules) then the simulator can discern which module was performing out of specification.

3.5 Ongoing CLOUDES Sound Implementation

As mentioned, we are implementing sound capabilities into the CLOUDES User Interface (UI) to evaluate the proposed research questions. We are currently evaluating using a scales-chords API provided by Scales-Chords.com (Scales-Chords 2017). The API provides chords and their corresponding visual representation to convey the notes that comprise each chord. Speaker icons (also provided through the Scales-Chords API) are clickable and play an audio file recording of the selected chord when clicked. Figure 3 provides a screenshot of the current view of the implementation using the *Basic* template model.

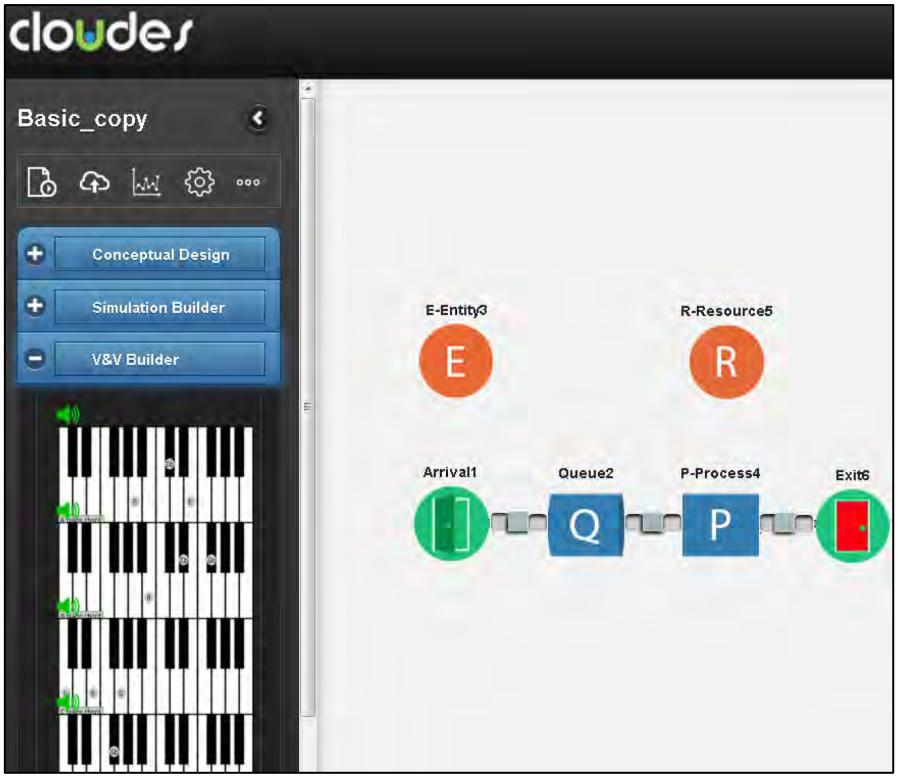

Figure 3: Example CLOUDES layout update to include sound options for the user with the speaker and keyboard images come from www.scales-chords.com/api.

4 DISCUSSION

We posit that modelers face similar challenges with incorporating sound cues into their simulations as the challenges faced when adding visual components. Vernon-Bido, Collins, and Sokolowski (2015) identify several challenge areas of visualization when designing models, including misrepresentation of model components, misrepresentation of the magnitude of changes by cutting off bounds, and adding complexity due to poor visualization. These challenges apply to sound cues in the following ways:

- *Challenge with misrepresenting model components*: Assigning the same or very similar sounds to different types of modules within a simulation will lead to confusion in discerning which modules are moving out of their acceptable ranges.
- *Challenge with misrepresenting the magnitude of model changes*: Altering the pitch of a sound based on local min/max will provide the effect that a minimum and maximum value always occur during execution. Additionally, the highest and lowest pitch values could occur multiple times as local minimum and maximum values can occur multiple times throughout execution. However, altering pitch based on the minimum and maximum values based on the model's specifications will only allow the pitch to reach its highest or lowest values if the minimum and maximum values are actually reached during execution.
- *Challenge with added model complexity*: competing sounds due to multiple of the same module (such as a model consisting of multiple Queues) using the same sound cues increases the difficulty of the model user being able to easily identify which module is operating outside of its Acceptable Range. Care must be taken to setup the simulation in a manner that facilitates and does not hinder this identification process.

Liu et al. (2014) identify technical challenges surrounding visualization which are also of importance to the use of sound cues in simulations, including usability, scalability, and integrated analysis of heterogeneous data. Usability refers to the success in designing visualizations that successful contribute to the advancement and use of visualization research. Scalability refers to how well the tools can contribute to large data sets. Integrated analysis of heterogeneous data deals with visualizing data obtained from various locations in various formats. These challenges apply to the use of sound cues in the following ways:

- Usability: the controls that allow modelers to incorporate sound cues into their models need to clearly convey their intent. The responsibility of understanding and conveying to the users of the simulations what the sounds represent within the simulation is the responsibility of the modeler.
- Scalability: the ability to simultaneously generate sounds for various modules can lead to confusion about which module is operating outside of its acceptable range, which decision paths are being taken, and hearing the sound cues for entities being separated over top of large numbers of Process and Queue modules operating within their normal ranges.
- Integrated analysis of heterogeneous data: the process for assigning sound cues for different types of data (i.e. discrete values, Boolean values, or percentages) needs to be clear for the modeler.

However, these visualization and technical limitation challenges are areas where sound can become very useful. Model complexity, for instance, could benefit from sound targeted to specific areas of the model instead of the model as a whole. As in any modeling initiative, the challenge is how we use the tools appropriately and it is in this appropriateness where sound and simulation research is promising.

Incorporating sounds into game-based simulation for teaching can enhance the sensory stimuli of the game and further connect the simulation user to the real system to facilitate learning. Games for training users can make use of the real sounds that occur within the real system to enhance realism at appropriate points during execution. The user can then use the sound cue as an opportunity to examine the current state of the simulation and gain insight into the conditions that caused the event associated with the sound to occur. Additionally, sound cue research provides a step towards the creation of methods for assisting visually impaired individuals in building and executing simulations similar to programming languages built for blind users (Sánchez and Aguayo 2005).

Sound can extend simulation usage to learners, modelers, and educators of the visually impaired. Tools, such as the Textual and Graphical User Interfaces for Blind People, exist to enable blind users to use speech-based mechanisms to inform the user of what is on the computer screen and help the user navigate the screen (Crispien and Petri 1994). Blenkhorn and Evans (1998) present the use of talking tactile diagrams to enable blind users to create, read, and edit flow diagrams. Teaching environments using 3-D sound simulation has been used with blind children to improve their identification, localization, and tracking skills (Inman, Loge, and Cram 2000). Applications and tools such as these can further connect users with simulation systems and can exist as implementation options alongside the addition of sound features within simulation environments.

Adding sound cues to a simulation will likely increase complexity in designing and implementing models due to the modeler needing to understand where it is appropriate to use sounds, what types of sounds are appropriate for the system being modeled, and the types of sounds that are applicable for the purpose of the model. This increased burden will then fall onto the users of simulation tools that provide sound cue capabilities. Using and creating simulations facilitates teaching and learning complex concepts. According to Robinson (2005), "although software vendors provide training in the use of their packages, in general they do not train their users to be simulation modellers" (p. 625). Ultimately, we want M&S students and professionals trained to understand when to use sound appropriately and not just how to use a tool.

5 CONCLUSION

This paper serves as a conceptual proposal for the addition of sound to designing and running simulations. In this regard, we have yet not tested the efficacy of having a simulation with sound against that of one without, as that falls outside the scope of this paper. While this effort is currently conceptual and formal studies on the effectiveness of sound within simulations have not yet been explored, we are developing the tool capability for its testing. The aim of adding sound cues to CLOUDES is to explore the ability to create a more intuitive modeling UI for model builders. Ultimately, we seek to expand the capabilities of M&S in general, not just the DES paradigm. By having more avenues in which to critique their models, sound can increase modelers' abilities to identify necessary model updates. Both Rauterberg and Styger (2005) and Gaver et al. (1991) conclude that their simulators improved both understanding and usage of their respective models, showing that sound can benefit modelers' responses to their simulations.

We present an initial extension to CLOUDES to add a sensory dimension to the creation and execution of simulations. Our initial implementation using third party sound APIs shows promise in incorporating our proposed ideas into an existing simulation environment. Adding sound helps to draw the modeler into the simulation so that changes can be heard over time and irregularities in the sound can assist the modeler in recognizing potential errors. As both sound and visualization deal with sensory stimuli, we explore known challenges associated with visualizing results during runtime to illuminate potential challenges with using sound. These challenges reveal several paths forward for continuing this research, including: identifying how to avoid or overcome this issues; and evaluating the benefits that sounds provides over those of visualizations from a sensory perspective. We believe that the use of sound can greatly benefit the M&S community and this open many research avenues for using sound to connect M&S to additional user groups, such as the visually impaired.

REFERENCES

Balci, O. 1998. “Verification, Validation, and Testing.” In *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*, edited by J. Banks, 335-393. New York, NY: John Wiley and Sons, Inc.

Blenkhorn, P., and Evans, D. G. 1998. “Using Speech and Touch to Enable Blind People to Access Schematic Diagrams.” *Journal of Network and Computer Applications* 21(1): 17-29.

Bowen, A. P. 1999. “Using Games to Teach.” *Journal of Emergency Nursing*, 25(5): 415-416.

Bruner, G. C. 1990. “Music, Mood, and Marketing.” *Journal of Marketing* 54(4): 94-104.

Crispien, K., and Petrie, H. 1994. “The'GUIB' Spatial Auditory Display-generation of an Audio-based Interface for Blind Computer Users.” In *Proceedings of the 1994 International Conference on Auditory Display*. Atlanta, GA: Georgia Institute of Technology.

Fortmann-Roe, S. 2014. “Insight Maker: A General-purpose Tool for Web-based Modeling & Simulation.” *Simulation Modelling Practice and Theory* 47: 28-45.

Gaver, W., Smith, R.B., and O’Shea, T. 1991. “Effective Sounds in Complex Systems: The Arkola Simulation”. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 85-90. New York, NY: ACM.

Hall, J., and Lawson, R. A. 2008. “Using Music to Teach Microeconomics.” *Perspectives in Economic Education Research* 4(1): 23-36.

Inman, D. P., Loge, K., and Cram, A. 2000. “Teaching Orientation and Mobility Skills to Blind Children using Computer Generated 3D Sound Environments.” In *Proceedings of the 2000 International Conference on Auditory Display*. Atlanta, GA: Georgia Institute of Technology.

Levy, D. L., and Byrd, D. C. 2011. “Why Can't We Be Friends? Using Music to Teach Social Justice.” *Journal of the Scholarship of Teaching and Learning* 11(2): 64-75.

Liu, S., Cui, W., Wu, Y., and Liu, M. 2014. “A Survey on Information Visualization: Recent Advances and Challenges.” *The Visual Computer* 30(12): 1373-1393.

Moore, E. B., Chamberlain, J. M., Parson, R., and Perkins, K. K. 2014. “PhET Interactive Simulations: Transformative Tools for Teaching Chemistry.” *Journal of Chemical Education* 91(8): 1191–1197.

Padilla, J., Diallo, S., Barraco, A., Lynch, C. J., and Kavak, H. 2014. “Cloud-based Simulators: Making Simulations Accessible to Non-Experts and Experts Alike.” In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. Diallo, I. Ryzhov, L. Yilmaz, S. Buckley, and J. Miller, 3630-3639. Piscataway, NY: IEEE Press.

Padilla, J. J., Lynch, C. J., Kavak, H., Diallo, S. Y., Gore, R., Barraco, A., and Jenkins, B. 2016. “Using Simulation Games for Teaching and Learning Discrete-Event Simulation.” In *Proceedings of the 2016 Winter Simulation Conference*, edited by T. M. K. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. E. Chick, 3375-3384. Piscataway, NJ: IEEE Press.

Rauterberg, M., and Styger, E. 2005. “Positive Effects of Sound Feedback during the Operation of a Plant Simulator”. In *Human Computer Interaction*, edited by B. Blumenthal, J. Gornostaev, and C. Unger, Lecture Notes in Computer Science, 876: 35-44, Berlin.

Robinson, S. 2005. “Discrete-event Simulation: from the Pioneers to the Present, What Next?” *Journal of the Operational Research Society* 56(6): 619–629.

Rohrer, M. W. 2000. “Seeing is Believing: The Importance of Visualization in Manufacturing Simulation.” In *Proceedings of the 2000 Winter Simulation Conference*, edited by J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, 1211-1216. Piscataway, NJ: IEEE.

Sánchez, J., and Aguayo, F. 2005. “Blind Learners Programming through Audio.” In *CHI'05 Extended Abstracts on Human Factors in Computing Systems*, 1769-1772. New York, NY: ACM.

Sargent, R. G. 2013. “Verification and Validation of Simulation Models.” *Journal of Simulation* 7(1): 12-24.

Scales-Chords. 2017. “Scales-Chords API for Webmasters.” https://www.scales-chords.com/api/.

Stratton, V. N., and Zalanowski, A. H. 1991. "The Effects of Music and Cognition on Mood." *Psychology of Music* 19(2): 121-127.

Tinari, F. D., and Khandke, K. 2000. "From Rhythm and Blues to Broadway: Using Music to Teach Economics." *The Journal of Economic Education* 31(3): 253-270.

Vernon-Bido, D., Collins, A., and Sokolowski, J. 2015. "Effective Visualization in Modeling & Simulation." In *Proceedings of the 48th Spring Simulation Multi-Conference - Annual Simulation Symposium*, 33-40. Vista, CA: Society for Computer Simulation International.

Wenzel, S., Bernhard, J., and Jessen, U. 2003. "Visualization for Modeling and Simulation: A Taxonomy of Visualization Techniques for Simulation in Production and Logistics." In *Proceedings of the 2003 Winter Simulation Conference*, edited by S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, 729-736. Piscataway, NJ: IEEE Press.

Xiang, X., Kennedy, R., Madey, G., and Cabaniss, S. 2005. "Verification and Validation of Agent-based Scientific Simulation Models." In *Proceedings of the 2005 Spring Simulation Multi-Conference - Agent-Directed Simulation*, 47-55. Vista, CA: Society for Computer Simulation International.

AUTHOR BIOGRAPHIES

JUSTIN DEURO is a Masters of Engineering student at Old Dominion University's Department of Modeling, Simulation, and Visualization Engineering. His email address is jdeur001@odu.edu.

CHRISTOPHER J. LYNCH is a Senior Project Scientist at the Virginia Modeling, Analysis and Simulation Center. He is currently pursuing a Ph.D. in M&S from ODU where he also received his MS in M&S in 2012 and a BS in Electrical Engineering in 2011. His research interests include multi-paradigm modeling, conceptual modeling, and verification of simulation models. His email address is cjlynch@odu.edu and his web page is http://www.odu.edu/directory/people/c/clync008.

HAMDI KAVAK is a Ph.D. candidate in the Modeling, Simulation, and Visualization Engineering Department at Old Dominion University (ODU). He received his master's degree in Modeling and Simulation from ODU. His research focuses on data-driven agent-based simulations with an emphasis on human mobility and cyber security. His email address is hkava001@odu.edu and his webpage is http://www.hamdikavak.com.

JOSE J. PADILLA is Research Assistant Professor at VMASC at ODU. He leads the Simulation Environments Lab at the Center. His research focuses on the understanding and modeling of problem situations. He received his Ph.D. in Engineering Management from Old Dominion University. His email address is jpadilla@odu.edu and his webpage is https://www.odu.edu/directory/people/j/jpadilla.