TOWARDS AN IAMU SIMULATOR SUITE: SIMULATION OF A BURNER ASSEMBLY REPLACEMENT AND LINE SNAPBACK USING UNITY3D GAME-MAKING SOFTWARE

[1]PEREZ-GRUSZKIEWICZ E. SERGIO, [2]KUMAR SHASHI, [3]BAUMGART ELWOOD, [4]RYAN JOHN, [5]LINDMAN WILLIAM

[1,2,3,4,5] *U.S. Merchant Marine Academy, USA*

ABSTRACT

In this work we use Unity3D game-making software for two simulations: the replacement of a fuel burner assembly on a marine boiler and line snapback. The burner assembly simulation is unique in that it requires a considerable amount of physical manipulation – in addition to shutting off the flow of steam and fuel, the clamping mechanisms holding the assembly must be unscrewed and manipulated downward, and the burner must be pulled out and properly oriented. The line snapback simulation will enable the student to set-up mooring and tow lines in a variety of configurations, and to see the effects of these configurations on the likelihood of line fracture/snapback. Our long-term goal is to build a suite of open-source training simulations in collaboration with other institutions.

Keywords: *Simulator, burner assembly, line snapback, game-making software, Unity3D.*

1.   INTRODUCTION

At the 2012 IAMU conference we presented our simulator for the start-up of a steam turbine, made with Unity3D game-making software. The goal of the project was to demonstrate that Unity3D could be used to make realistic and useful PC-based simulators "from the ground-up".

At that conference we found some interest in the idea of IAMU institutions joining in making simulators for the training of professional mariners. Probably the most feasible way to do this is for institutions to make simulators independently, and to then join the simulators to make a suite of training tools. Another possible method of collaboration is for some institutions to create the 3-D (3-dimensional) artwork required and others to use the art-work for the creation and programming of the simulation.

For the upcoming IAMU conference we chose to create simulations for the replacement of a fuel-oil burner assembly, and for the factors influencing line snap-back. We hope to gain STCW approval for the burner assembly simulation, which would relieve us of some of the staffing problems caused by administering tests in this competency.

At the onset of the project we were uncertain if a task requiring as much mechanical manipulation as the changing of a burner assembly could be effectively simulated, but we have come to conclude that the simulation is an effective one that would well-train students for this task, and help them pass the STCW assessment.

The line snap-back simulation is to help train our sailors in this very important safety issue, with the goal of eliminating injuries. At the time of this writing the simulation is not completed, but we hope to do so by the time of the IAMU meeting and present our results there.

The making of any simulator requires two separate tasks: the creation of the 3-D artwork (for example, for the line snapback simulation we needed models of a tugboat and line-handling hardware), and the programming of the simulation (that is, to enable the user to click on 3-D objects and obtain a desired result, or to move around in 3-D space). In the following sections we will describe how we completed each of these tasks.

2.   3-D ARTWORK

There are numerous software products available for the creation of 3-D artwork. Products such as Maya and Studio 3DS Max cost thousands of dollars to purchase; in the right hands these products can produce 3D art of stunning realism. Other products such as Silo cost only about $120 per copy, and are still quite powerful. The freeware Blender can also produce beautiful and realistic 3D artwork, but has a steeper learning curve than the other products; in addition, the documentation is not as thorough as with the commercial products.

The choice of Studio 3DS Max was an easy one to make for this project, as the USMMA has a license for this product. This software was very easy to learn, and the documentation extensive. There are countless videos on YouTube detailing how to make almost any shape.

We estimate that the artwork for the burner assembly simulation was done in about 100 hours of work, much of which involved learning how to use the software. An experienced artist could have performed the same work in a fraction of the time – about 10 hours or less, and probably achieve a better result. This is the reason why most game-making studios have dedicated 3D artists on their staff.

The burner assembly artwork can be seen in figure 1, and consists of the burner assembly, the clamps holding the assembly in place, the pipes and valves, and the boiler itself. The warehouse-like building was purchased on the internet for about $20, and included the plastic boxes and storage shelf visible in the scenes.
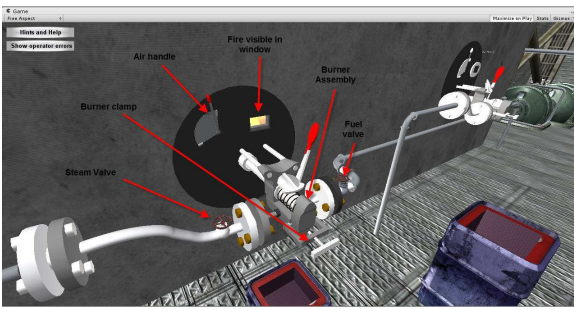
Figure 1 Burner assembly scene. These 3D objects were created with Studio3DS Max or purchased on the internet.

Figure 2 shows the 3D model of the tugboat used for the line-snapback simulation. This model was purchased from the internet for about $150. Some modifications with Studio 3DS Max will be necessary to the towing equipment on the aft deck in order to simulate a range of line tying scenarios. The money spent on this model was justified, as the authors of this paper could not have created artwork this realistic in a reasonable amount of time.

Figure 2 The tugboat model purchased for the line snapback simulation.

3. PROGRAMMING

Once the 3D artwork was available the simulation could be programmed. Both simulations were to be used as training and evaluation tools requiring no intervention from instructors – even the assessment of student performance was required to be automatically done by the software. The inordinate amount of time which would otherwise have to be spent performing the training and STCW assessment of the burner assembly replacement was a major driving force in the selection of the burner assembly for simulation.

Unity3D is arguably the most powerful game-making software product available on the market. As detailed in our previous work, Unity requires programming in one of two languages: Java or C#. We selected the Java platform because it is somewhat simpler and we were more familiar with the Java syntax.

The user of the burner assembly simulator must first select the proper safety gear from a list of protective clothing, otherwise a warning message will be created which will be visible to both the instructor and the user. Figure 3 shows the menu for this feature.
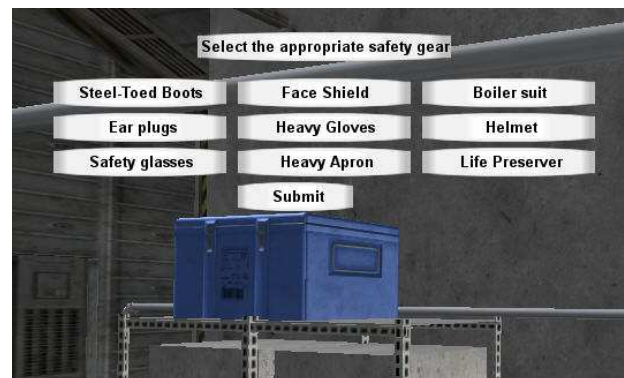

Figure 3 The user of the simulator must select the appropriate safety gear from the menu.

After selecting the proper protective gear, the user must close the fuel, atomizing steam and air valves. The proper sequence must be followed or a warning message will be displayed. Clicking on the valves makes a menu appear which permits opening and closing the valves as well as displaying the steam or fuel pressure. Please see Figure 4.
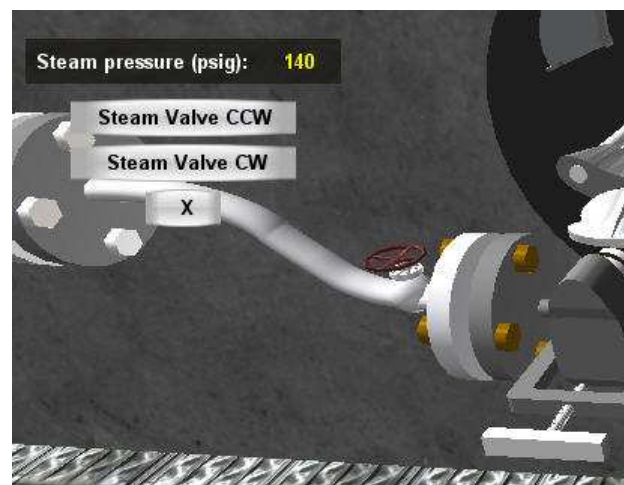

Figure 4 A menu appears when the steam or fuel valves are clicked, and the pressure displayed.

Once the valves have been opened, the user can remove the clamps which hold the burner assembly in place. This is accomplished by clicking on the clamp assembly. The screw holding the clamp must first be loosened by clicking on its menu item, after which the clamp can be moved down. Figure 5 shows the screw and clamp after moving into a position which will allow removal of the burner assembly.
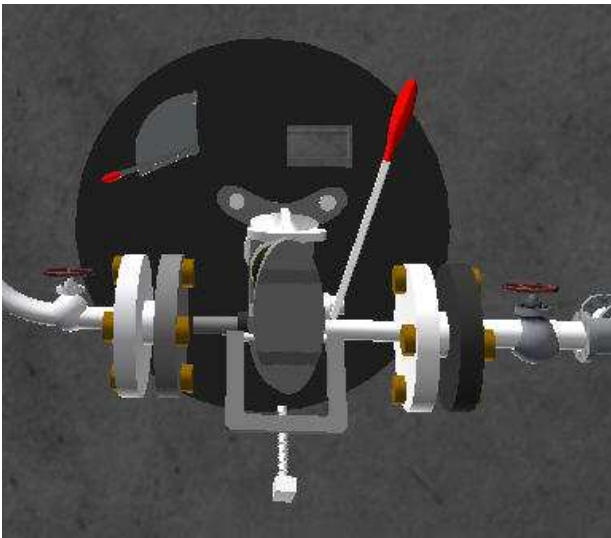
Figure 5 Clamp in the down position, allowing removal of the burner assembly

If the user neglects to shut off the oil or steam before loosening the clamp, the simulation will spray oil and steam on the scene. In this way, the simulator is superior to learning on the actual equipment, as exposing the student to spraying steam and oil could be impractical as well as unsafe. Figure 6 shows the spraying oil and steam.

Figure 6 Steam and oil spray if the user neglects to turn off the flow of fuel and steam before loosening the burner clamp.

If the user closes the valves properly, he or she will be able to click on the burner in order to make appear the burner menu, which will allow the user to move the burner assembly out of the boiler, as shown in figure 7.
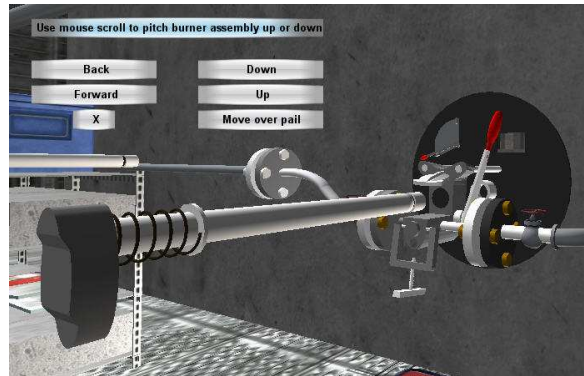

Figure 7 The burner can be moved using the burner menu. Here the burner has been moved out of the boiler, and is now ready to be placed in an appropriate container. Note the replacement burner in the background

Before the burner is moved it should be pitched up to prevent oil leaking on the deck, as shown in figure 8. Not doing so causes a warning message to appear. The burner is pitched up or down using the mouse scroll wheel.
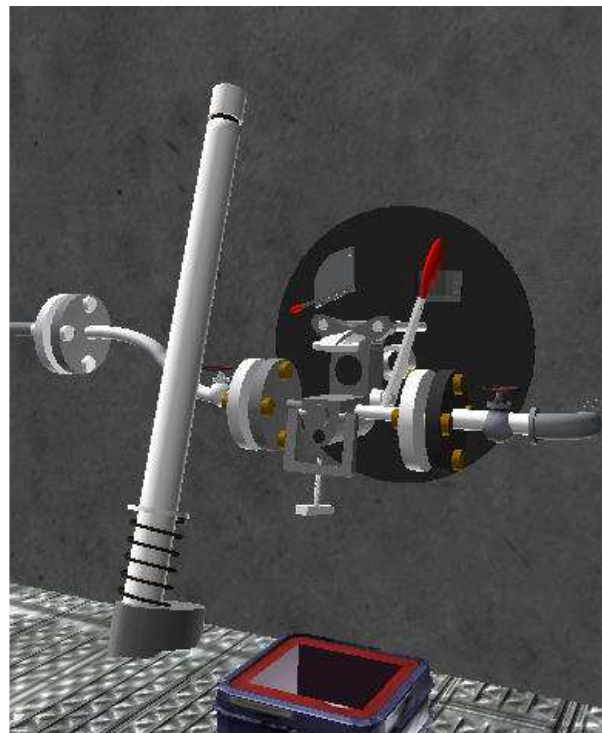

Figure 8 The burner assembly pitched up to prevent spillage of oil on the deck.

Once the burner has been pitched up, it can be moved over the storage container, pitched down, and then placed into the container, as shown in Figure 9.
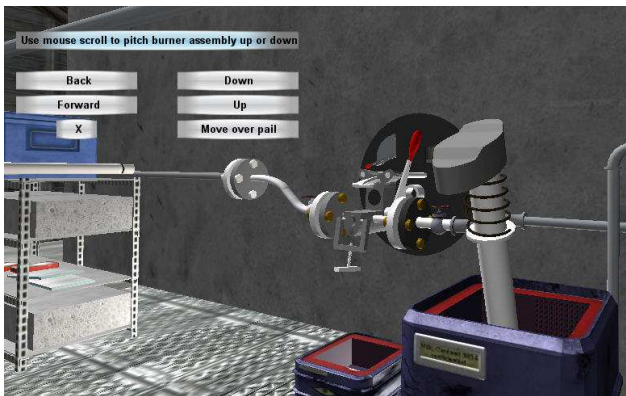
Figure 9 The burner assembly placed in its appropriate container

Finally, the new burner assembly can be placed into position and moved into the boiler. The steam, air and fuel valves then must be opened in the proper sequence to prevent a warning message from being displayed, and the flame is ignited automatically. Figure 10 shows the new burner assembly before moving.
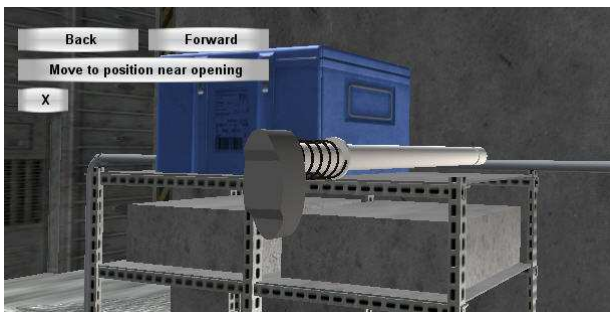

Figure 10 The replacement burner assembly before moving into the boiler.

4.   CONCLUSIONS

We intend to submit the burner assembly simulation to the U.S. Coast Guard for approval to enable USMMA to use the simulation for STCW assessment. This will alleviate staffing problems caused by the time-consuming process of performing the assessment of this competency.

At the time of this writing the simulation has not been tested on students in order to determine the effectiveness of the simulation, but it appears that the burner assembly simulation can be a useful tool for the training and assessment of mariners. One anecdote has been offered by the programmer of the simulation, whose 10-year old daughter has learned from the simulation how to replace a burner. We expect to have some data on the effectiveness of the simulation by the time of the next IAMU conference.

The creation of a simulator of this type is not an overly onerous task, but it does require some time and dedication. Creating the artwork can be particularly time-consuming, especially for the inexperienced. Often artwork can be purchased on the internet which can greatly reduce the time and effort required. The burner artwork was created in about 100 hours, and the programming also done in about 100 hours.

The programming for this simulation was not very difficult. The Java language used by the Unity game engine is easy to work with, and the programming process has been made as simple as possible by the creators of Unity. Other more complex simulations such as bridge simulators or full engine room simulators would certainly be within the capabilities of Unity3D, but would be considerably more difficult to create.

Our final goal is to stimulate and join other IAMU members in creating a suite of simulators which can be used by all marine institutions free-of-charge.

5.   ACKNOWLEDGMENTS