Applied Information Science Research in a Virtual World Simulation to Support Robot-Mediated Interaction Following the Fukushima Nuclear Disaster

Michael Vallance¹, Catherine Naamani², Malcolm Thomas³, James Thomas⁴

¹Department of Media Architecture, Future University Hakodate, Japan
²The Centre for Excellence in Learning and Teaching, University of South Wales, Pontypridd, South Wales, UK
³Mountain Ash Comprehensive School, Rhondda Cynon Taf, South Wales, UK
⁴Aberdare High School, Rhondda Cynon Taf, South Wales, UK

¹michael@fun.ac.jp; ²catherine.naamani@southwales.ac.uk; ³Malcolm_Thomas@macs.rhondda.sch.uk; ⁴jthomas@aberdarehighschool.co.uk

Abstract - The absence of robots to assist with the repair efforts in the immediate aftermath of the Fukushima Daiichi power plant disaster of March 2011 revealed much about Japan’s lack of preparedness for nuclear accidents. The Fukushima Nuclear Accident Independent Investigation Commission highlighted insufficient knowledge and training at the plant, with residents being left confused by the conflicting information regarding the dangers of the reactor explosions. The research summarized in this paper examines how students in Japan and UK collaborate towards the development of a better understanding of the challenges and possible solutions when dealing with disaster recovery such as Fukushima. The context for collaboration is set within a 3D virtual space and Fukushima simulation where students program LEGO robots to follow distinct circuits. The international collaboration by students as non-experts has highlighted the benefits and challenges posed when engaged in constructing robot-mediated interactions (RMI) within 3D virtual simulations. Students’ immersion (or flow), Circuit Task Complexity and Robot Task Complexity have been collated to create a new metric for tasks involving robots, which we have termed Task Fidelity.

Keywords- LEGO Robots; Virtual Worlds; Collaboration; Information Science; Task Fidelity; Japan; Wales

I. BACKGROUND

On March 11, 2011 a large earthquake and tsunami badly damaged cooling systems to reactors at the Fukushima Daiichi nuclear plant in Japan. Four reactors exploded and radioactivity was released to the atmosphere. Residents within 20km were evacuated, and many Japanese and non-Japanese as far away as Tokyo (220km from Fukushima) decided to ‘evacuate’ to Osaka and overseas. In December 2011 the reactors were finally declared stable but the effects continue to exist: evacuees cannot return home and depression is becoming prevalent among the strained residents [1]; the Japanese government has changed its criteria for dangerous levels of radioactivity so leaving residents confused [2]; workers are struggling to maintain the safety of the plant [3]; deformities have been discovered in local wildlife [4]. In July 2012 a Japanese parliamentary panel, The Fukushima Nuclear Accident Independent Investigation Commission, produced a comprehensive report highlighting the collusion by the power plant company (TEPCO) and government regulators (NISA), insufficient knowledge and training, lack of genuine preparedness, and an insular attitude of ignoring international safety standards. The report concluded that the accident was not a ‘natural’ disaster but was “a profoundly man-made disaster” [5]. Because the mindset that allowed the accident to happen can be found across the country, the panel called the disaster ‘Made in Japan’. The BBC has filmed an informative documentary of the days following the disaster entitled ‘Inside the Meltdown’.

One of the most surprising technology related episodes during the post-disaster efforts was Japan’s lack of robots to assist with the recovery operations. Japan, a robotics-friendly nation with the world’s highest levels of automation, had to count on foreign assistance. Less than a week after the tsunami and earthquake, iRobot USA donated two PackBot 510 robots with hazmat kits and two Warrior 710 robots with manipulator arms. These robots were not ‘Made in Japan’. iRobot engineers trained Japanese operators the following week yet it took three weeks for TEPCO to authorize their use [6]. An anonymous worker blogged daily about his experiences using the donated iRobots to assist efforts to stabilize the damaged reactors. He highlighted, often with tragic humor, the barriers to working effectively: inept supervisors, demanding schedules, lack of resources, poor communication infrastructure, neglect of safety, supervisors’ instructions to ignore dosimeter alarms, lack of coordination, and delays in deploying robots. The blog was extremely informative but in August 2011 it completely disappeared. It’s unclear whether TEPCO or the blogger’s supervisors demanded that the material be removed. However, IEEE has archived and translated the blog in its entirety and remains a fascinating, pragmatic insight to the complexities of post-disaster recovery.
[6]. The Fukushima Daiichi nuclear power plant disaster revealed much about Japan’s lack of preparedness for nuclear accidents. Despite the brave efforts of its labor force leading up to, and in the aftermath of, the reactor explosions, it became apparent that coordination and communication were disorganized.

II. MOTIVATION

As Information Science educators, two specific observations can be made from the multitude of issues surrounding these tragic events:

1. People need to be better informed and also equipped to make sense and meaning of information. In education contexts, we should not tell students what to think but to give them learning opportunities for reflecting, organizing, negotiating, and creating. An interesting but challenging project like programming robots provides opportunities for learning content in the Science, Technology, Engineering, and Math (STEM) subjects in context.

2. International collaboration is essential communication for now and the future. A virtual world as a future 3D space provides a safe medium for communication, collaboration, and experiential learning.
It is posited that Higher Education academics, researchers and school educators need to find ways to support experience of collaboration, and provide awareness of the challenges of nuclear accident recovery. The Virtual Collaborative Spaces research summarized in this paper has developed a 3D simulation of the Fukushima nuclear plant in a Unity3D virtual space (see Fig. 1). Prior to entering the nuclear plant simulation undergraduate students in Japan studying Media Architecture (N=6) and A-level students in UK studying science-based subjects (N=10) have been undertaking robot related tasks in a learner-designed, OpenSim 3D virtual space (see Fig. 2).

Information Science education is a multi-discipline endeavor. Information Science research should therefore incorporate elements of computer science, cognitive science, the social sciences, communication and design. Consequently, the purpose of this paper is to: (i) summarize the iterative design and technology developments employed by the researchers and the students; and (ii) outline educational value as students engage in collaboratively constructing robots in the 3D virtual world. The paper will suggest a new metric for measuring robot tasks, which we term Task Fidelity.

III. PREVIOUS RESEARCH

Learning is considered to be a “process whereby knowledge is created through the transformation of experience” [7]. deFreitas and Neumann suggest that the appeal, immersivity, and immediacy of virtual worlds can support this ‘experiential learning’ but education requires a re-consideration of how, what, when, and where we learn [8]. deFreitas and Neumann use Dewey’s [9] concept of inquiry (pre-reflection, reflection, and post-reflection) to posit that learners’ virtual experiences, their use of multiple media, the transactions and activities between peers, and the facilitation of learner control between them will lead to ‘transactional learning’ which “aims to support deeper reflection upon the practices of learning and teaching” which arguably leads to “wider opportunities for experiential learning” [8]. It is posited that knowledge is not merely a commodity to be transmitted, encoded and retained, but is personal experience to be actively constructed [10].

Our applied research is developing an evidence-based framework of learning within tasks of measurable complexity in virtual worlds. This is currently undertaken by collecting data of collaboration in a virtual world: (i) capturing students’ procedural processes as they work through the task solution; and (ii) capturing students’ learning reflections during and after completion of tasks. From this data we then associate specific cognitive instances within the task process to a taxonomy of cognition [11]. It is posited that this will lead to a better understanding of tasks performed in a virtual world in which we can expect specific learning to occur.

Our research has been designed to collate data of students collaborating in-world when programming a robot to follow pre-determined mazes. The students’ aim is to effectively communicate solutions to problems which involve the programming of a LEGO robot to follow the specified circuits. This is undertaken by (i) designing circuits which necessitate the use of robot maneuvers and sensors; and (ii) experiencing collaboration in a virtual world with students in Japan and UK. These experiences lead to personal strategies for teamwork, planning, organizing, applying, analyzing, creating and reflection. Complex problems are thus presented which necessitate the use of programming skills, collaboration, and the aforementioned cognitive experiences.

Each maze task represents tangible and quantifiably measured outcomes. The robot selected for the programming tasks was LEGO robot 8527 supported by the LEGO Mindstorms NXT software version 2.1 and LabView 2010 with NXT module. LEGO robot 8527 was adopted due to its simplicity and potential for sensors to be added as the research tasks developed.

In order to establish objectives and a focus for measuring task success and learning, students were asked to design and complete a number of tasks of varying complexity. The tasks were to design two-dimensional mazes for their robot to successfully navigate from the start to completion in the physical and virtual environments. In order to quantify each task complexity the programming of the LEGO robot required a determination of an action and a vector.

Our previous research, reported in Vallance and Martin [12], found that there is no consensus in the discipline of Robotics or Human-Robot Interaction for accurately measuring task complexity. Given the specific purposes of the robot in our research, we utilized the eminent work in robotics by Barker and Ansorge [13] and also Olson and Goodrich [14]; where task complexity is calculated according to the number of sections that make up a given maze. We called this Circuit Task Complexity (CTC) which equals the number of directions + number of maneuvers + number of sensors + number of obstacles.

\[ CTC = \sum (d + m + s + o) \]

CTC = \sum (4 + 3 + 2 + 2) = 11

For example, in Fig. 3 the robot must maneuver around at least 2 obstacles in order to reach its target. The number of directions to be programmed are 4, the number of maneuvers are 3, and the number of sensors is 2 (i.e. two touch sensors).

\[ CTC = \sum (d + m + s + o) \]

- 224 -
Given the steadily rising level of Circuit Task Complexity and students’ increasing mastery of the more challenging tasks as evidenced by their ability to complete them with fewer errors and in less time, a taxonomy of cognition developed by Anderson et al. [11], and used as our research instrument, suggests that some developmental pattern should be expected to emerge as the procedural knowledge required to complete the tasks came to be more effectively applied and as student accomplishment increased. However, in over 60 hours of data collected and analyzed it was surprising to see no consistent development in the frequency with which particular cognition appeared over time. For example, the relative frequency of ‘applying procedural knowledge’ was not patterned in an upward trend as tasks progressed and difficulty increased. It was concluded that although virtual worlds can be positioned to construct new learning environments with unique tools and communication opportunities, our research did not reveal measurable evidence of higher-order thinking and learning as task complexity increased [12].

IV. CURRENT RESEARCH

Task complexity was initially calculated according to the number of sections that made up a given circuit, where a section was defined as an element that was different in orientation (direction) to the preceding section. We called it Circuit Task Complexity (CTC). However, we found that the logic of assigning task complexity to circuits was inadequate. For instance, initially we assigned complexity values to distinct maneuvers such as forward – turn – back. We found over the course of our previous research that as circuits became more challenging, the NXT programming became more complex. This was especially the case when we needed to add sensors to maneuver around and over obstacles. Simply adding the number of obstacles to the circuit task complexity was flawed because the programming required to maneuver over a bridge using touch sensors, for instance, was far more complex than maneuvering around a box using touch sensors. Consequently, we modified our task complexity to be determined by the NXT program solution rather than the circuit to be navigated. We call this Robot Task Complexity (RTC), which is measured as:

\[
RTC = \sum M v_1 + \sum S v_2 + \sum SW + \sum L v_3
\]

where,

\[M = \text{number of moves (direction and turn)}\]
\[S = \text{number of sensors}\]
\[SW = \text{number of switches}\]
\[L = \text{number of loops}\]

where \(v = \text{number of decisions required by user for each programmable block}\)

\[v_1 = 6\]
\[v_2 = 5\]
\[v_3 = 2\]
In the NXT Mindstorms software, the Move block controls the direction and turns that the LEGO robot will take. There are six variables that need to be considered: NXT ‘brick’ port link, direction, steering, power, duration, and next action. In other words, the students have to make six specific decisions about the values which make up the programmable block. Therefore, we assign $v_1$ a value of 6. There are eight common sensors which are used in our tasks (timer, light, ultrasonic, color, touch, sound, distance, wait) with the sensors’ capabilities determined by 5 variables (so we assign $v_2 = 5$). Although some sensors have 6 decisions built in and some have 5, the difference is that the extra decision is simply cosmetic as in ‘speak an alert’ so does not impact on the robot’s performance or capability to complete the task. All sensors are tagged as $S$. A loop has only two variables to consider so we assign $v_3 = 2$.

Given the circuit shown in Fig. 3 above, the robot has to be programmed to move in 4 directions, with 3 turns and 2 touch sensors. ANXT program solution in Fig. 4 can then be used to calculate the Robot Task Complexity.

$$RTC = \sum M v_1 + \sum S v_2 + \sum SW + \sum LV_3$$

$$RTC = (8 \times 6) + (3 \times 5) + 0 + 3$$

$$RTC = 66$$

We acknowledge that, at present, our modified Robot Task Complexity metric applies only to the Mindstorms NXT software and LEGO robot, but it does provide a useful indicator in our attempts to analyze the experiential learning during the collaborative tasks. The CTC problem can now be evaluated against the RTC solution. This will be discussed in Section VI.

V. IMPLEMENTATION – DESIGN & TECHNOLOGY

Open Simulator (commonly referred to as OpenSim, an open source multi-platform, multi-user 3D application server) was chosen and Reaction Grid selected as the service for locating the virtual space. The learner-centered design approach has enabled a number of innovative tools to be created and customized by the students: for instance, the ability to move a graphical representation of the LEGO robot object and leave a trace of the circuit in-world (see Fig 5.). Also, media objects in-world (known as prims) can display live streamed video from the lab in Japan (using the online UStream service) and also from an iPhone attached to the front of the LEGO robot (using the Bambuser app). Virtual noticeboards can be updated with text and images to aid communication. Although programming is mostly undertaken using the Mindstorms software, a LabView VI has been developed to enable communication directly between an in-world prim and the physical LEGO robot. A number of virtual and real world tools have been developed:

- LEGO NXT Mindstorms software and robot 8527 (constructed as per Task Complexity criteria).
- National Instrument’s robotSim 3D Simulator Robotics Starter Kit - Cogmation Robotics has been constructed and simultaneously programmed using LabView 2011 software.
- Real world obstacles can be represented in-world as prims that can be located to create the course for navigation.
- OpenSim viewers used are Firestorm and Impudence for Mac OSX. The OpenSim world is hosted on a remote Windows 2008 server.
• Video of one real-world location is streamed in-world. This is undertaken with QuickTime streaming using QuickTime Broadcaster and local Mac OSX server for viewing in-world the real-life robot construction. UStream is also used.

• We have an iPhone 5 attached to the front of a LEGO robot which allows simultaneous live video streaming of the robot’s movement via the Bambuser App.

• Telerobotic communication. A real-world robot is also controlled by a specially assigned, in-world object (or prim). A scripted prim in OpenSim relays a signal to a Visual Studio coded file on the Windows 2008 server hosting the virtual world. A LabView 2011 VI receives the signal from the server, acting as a Virtual Bridge, and relays it to the NXT controller (or ‘brick’) and LEGO robot. This in turn activates a LabView VI on a remote laptop which then sends a signal to the robot; for instance, activate a motor for wheel rotations. Currently we have the robot tethered via USB and are exploring Bluetooth communication.

The telerobotic communication allows students in UK to maneuver the robot in Japan via controllers in the virtual world. This technical development has mostly been undertaken through ‘trial and error’ and innovative construction by the students and metaverse designers at Reaction Grid (our virtual world hosting company).

A Virtual Bridge is a web application that listens out for http requests from virtual worlds and passes them on to a list of IP and port combinations where a TCP listener will be waiting for the data. This can be used to pass messages to robotics devices, such as a LEGO Mindstorms NXT controller.

Configuration for the Virtual Bridge application requires opening the file in the Visual Studio editor, scrolling down the file to the section marked <appSettings> and editing the values listed as IP and port in the format IP:Port,IP:Port,IP:Port. Any number of devices can be added to the list.

A Linden script (LSL) prepared in collaboration with Reaction Grid to listen for web requests on a given URL, and to send data to a specified http endpoint that can relay requests to external hardware devices via an in-world trigger (such as a touch event) is added to a prim in-world. It includes HTTP listener functionality to respond directly to the OpenSim server. The bridge is one-way and each time the script is compiled, a URL is generated to send requests back in-world to the prim.

In addition, a LabView VI (a program file) is set by default to listen for messages on port 2055. When a message is received from the server, the attached LEGO NXT robot motor is activated. A demonstration is available at our research website http://www.mvallance.net.

In addition we have created a virtual Fukushima nuclear plant using Unity3D-based technology called JIBE and is hosted by Reaction Grid at http://funi.jibemix.com/jibe/. In this space we can view a separate reactor plus the four cooling towers. We have a meeting space where we can upload images to a presentation board, and a glass structure for developing future learning spaces. We are currently programming a virtual robot to move within the Fukushima space and its maneuvers replicated in our real-world lab; as previously exemplified in our OpenSim simulation above. Video streaming from our real-world lab is currently displayed outside the virtual world but within a dedicated web browser window. Two major advantages of this Unity3D-based simulation is its ability to be accessed via a web browser and the space is relatively more naturalistic. We have conducted a number of ‘search and rescue’ tasks in our virtual Fukushima space.
VI. IMPLEMENTATION - TASKS

Recall that our current research is continuing to explore experiential learning with tasks of measurable complexity in 3D virtual worlds, and students’ tasks are evaluated based upon a modified Robot Task Complexity formula. Elliot et al. [15] argue that there is a need to better understand the affordances of 3D virtual worlds which maximize learning: “Immersivity is part of the learning experience and the context, from which pedagogy and cognitive affordances cannot be separated” (p.73). Given that our context is essentially Robot-Mediated Interaction (RMI) this applied research can thus determine how immersed students become within the process of each task. To record ‘immersion’ (a cognitive phenomena also referred to as ‘flow’ [16]), data are collected from the students during and after each task. At regular intervals during the task procedures each student has to answer two questions, with four options:

Q1. How challenging is the activity?
   Difficult (score 4) Demanding (score 3)
   Manageable (score 2) Easy (score 1)

Q2. How skilled are you at the activity?
   Hopeless (score 1) Reasonable (score 2)
   Competent (score 3) Masterful (score 4).

These questions were chosen based upon research in immersivity by Pearce et al. [17]; with optimal challenge – skill relationship the students become immersed in the Robot-Mediated Interaction (RMI) tasks. Immersion (or flow) is measured to determine if robots become more ‘human-like’ to the students as they remotely control, via the 3D space, a robot to complete the tasks. In optimal immersion scenarios the communication may then be interpreted as Human-Robot Interaction (HRI).

To date we have conducted and recorded reliable data from 28 tasks: (T1 to T21 and T28 were conducted in the OpenSim space and are discussed in section VII; T22 – T27 were conducted in the virtual Fukushima space and are discussed in Section VIII. We posit that this data can be used in conjunction with previous data of learning and communication [12] to develop a framework for virtual world learning. Some tasks have involved Japanese students collaborating with other remotely located Japanese students, and some with Japanese students and UK students [18]. Tasks have included maneuvering around obstacles using distance and turn commands, using touch sensors to find ways around obstacles, constructing a bridge and using touch sensors to move over obstacles, using light sensors to avoid obstacles, using RGB sensors to locate items, and manipulating the telerobotic controls to virtually maneuver our LEGO robot within the virtual Fukushima space as part of ‘search and rescue’ simulations [19].

<table>
<thead>
<tr>
<th>Task</th>
<th>Task description</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>Assemble LEGO robots. JPN + UK students introductions</td>
</tr>
<tr>
<td>T2</td>
<td>NXT program + circuit. JPN teaching UK</td>
</tr>
<tr>
<td>T3</td>
<td>NXT program + circuit (90 degree turns + measured length). UK teaching JPN</td>
</tr>
<tr>
<td>T4</td>
<td>Circuit + NXT program. Move. Touch sensor. Turn 90 degrees. JPN teaching JPN.</td>
</tr>
<tr>
<td>T5</td>
<td>Circuit + NXT program. Around obstacles. JPN teaching JPN.</td>
</tr>
<tr>
<td>T6</td>
<td>Circuit + NXT program. Around obstacles. JPN teaching JPN.</td>
</tr>
<tr>
<td>T7</td>
<td>NXT program + touch sensors + circuit. Locate and press switch off. JPN teaching JPN.</td>
</tr>
<tr>
<td>T8</td>
<td>Over an obstacle. NXT program + sensors + bridge building (cardboard). JPN teaching JPN.</td>
</tr>
<tr>
<td>T9</td>
<td>Over an obstacle. NXT program + sensors + bridge building (wood). JPN teaching JPN.</td>
</tr>
<tr>
<td>T10</td>
<td>Robot arm + scoop. UK teaching JPN</td>
</tr>
<tr>
<td>T11</td>
<td>Robot arm + NXT program. JPN preparation</td>
</tr>
<tr>
<td>T12</td>
<td>Robot arm + scoop + NXT program. Streaming video. JPN teaching UK.</td>
</tr>
<tr>
<td>T13</td>
<td>Programming LabView for remote control.</td>
</tr>
<tr>
<td>T14</td>
<td>Programming LabView for remote control.</td>
</tr>
<tr>
<td>T15</td>
<td>Programming LabView for remote control.</td>
</tr>
<tr>
<td>T16</td>
<td>UK teaching Japan. Robot construction + NXT program + stop and swing arm to hit ball.</td>
</tr>
<tr>
<td>T20</td>
<td>Robot construction + NXT program + obstacles + sensors.</td>
</tr>
<tr>
<td>T21</td>
<td>Suika robot. Rotate + follow line+ sensor + chop down. JPN teach UK.</td>
</tr>
<tr>
<td>T22</td>
<td>Programming LabView for remote control.</td>
</tr>
<tr>
<td>T23</td>
<td>Programming LabView for remote control.</td>
</tr>
<tr>
<td>T24</td>
<td>Remote control for search &amp; rescue circuit A.</td>
</tr>
<tr>
<td>T25</td>
<td>Remote control for search &amp; rescue circuit B.</td>
</tr>
<tr>
<td>T26</td>
<td>Remote control for search &amp; rescue circuit C.</td>
</tr>
<tr>
<td>T27</td>
<td>Remote control for search &amp; rescue circuit D.</td>
</tr>
<tr>
<td>T28</td>
<td>Move to black line, stop and throw ball to hit over obstacle. UK teaching Japan.</td>
</tr>
</tbody>
</table>
VII. RESULTS

The collated data include total challenge and skill values, Circuit Task Complexity values, and Robot Task Complexity values. In order to compare the data from Task 1 to Task 28 it was necessary to scale all the values between 0 and 1. For instance, for the challenge and skill values, in each task we simply divided the sum scores of the students by the maximum score possible. For the Circuit Task Complexity values we took the maximum CTC value and divided it into each CTC value. Similarly, for the Robot Task Complexity values we took the maximum RTC value and divided it into each RTC value. All values are thus represented between 0 and 1. This allows us to represent the data graphically and thus determine the immersion (or flow) in the case of challenge and skills, and Task Fidelity (TF) in the case of Circuit Task Complexity and Robot Task Complexity values. Incomplete task data was extracted from the analysis.

Pearce et al. [17] report two extremes if students find tasks too easy (bored) or too difficult (anxiety). They use measurements of skill and challenge in order to view an optimal path of immersivity (or flow); represented by the diagonal line 0,1 in Fig. 6. The graph shows that most tasks were of close proximity to the optimal flow path. According to the graph, students were fully immersed in Tasks 12, 18 and 21. These tasks involved the Japanese students preparing to teach the UK students (T18), and Japanese students teaching UK students (T12 and T21). Tasks 5, 11, 16 and 20 caused most anxiety. These tasks involved the Japanese students being taught by UK students (T16), and Japanese students teaching other Japanese students (T5). T11 and T20 involved preparation. The commonality of these four tasks (i.e. Tasks 5, 11, 16 and 20) though was the inclusion of obstacles thereby requiring the selection and programming of sensors. Looking at the task communication transcripts and screen captures it revealed that the students had to utilize different procedural knowledge when involving programming a touch sensor to coordinate with a motor action; thus indicating a more complex task. The complexity of the task can be quantified by calculating Task Fidelity. This is next discussed.

TABLE II IMMERSION DATA

<table>
<thead>
<tr>
<th>TASK</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>T5</th>
<th>T6</th>
<th>T7</th>
<th>T8</th>
<th>T9</th>
<th>T10</th>
<th>T11</th>
<th>T12</th>
<th>T16</th>
<th>T17</th>
<th>T18</th>
<th>T19</th>
<th>T20</th>
<th>T21</th>
<th>T28</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHALLENGE</td>
<td>0.5</td>
<td>0.75</td>
<td>0.5</td>
<td>1</td>
<td>0.8</td>
<td>0.67</td>
<td>0.67</td>
<td>0.42</td>
<td>0.42</td>
<td>0.8</td>
<td>0.58</td>
<td>0.8</td>
<td>0.25</td>
<td>0.7</td>
<td>0.25</td>
<td>0.75</td>
<td>0.75</td>
<td></td>
</tr>
<tr>
<td>SKILL</td>
<td>1</td>
<td>0.5</td>
<td>0.75</td>
<td>0.67</td>
<td>0.67</td>
<td>0.8</td>
<td>0.5</td>
<td>0.92</td>
<td>0.5</td>
<td>0.5</td>
<td>0.58</td>
<td>0.45</td>
<td>1</td>
<td>0.7</td>
<td>1</td>
<td>0.5</td>
<td>0.75</td>
<td>0.58</td>
</tr>
</tbody>
</table>

Fig. 6 Immersion of students in tasks

TABLE I TASK COMPLEXITY DATA

<table>
<thead>
<tr>
<th>TASK</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>T5</th>
<th>T6</th>
<th>T7</th>
<th>T8</th>
<th>T9</th>
<th>T10</th>
<th>T11</th>
<th>T12</th>
<th>T16</th>
<th>T17</th>
<th>T18</th>
<th>T19</th>
<th>T20</th>
<th>T21</th>
<th>T28</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTC</td>
<td>0.56</td>
<td>0.50</td>
<td>0.81</td>
<td>0.81</td>
<td>1.00</td>
<td>0.69</td>
<td>0.25</td>
<td>0.31</td>
<td>0.19</td>
<td>0.63</td>
<td>0.63</td>
<td>0.56</td>
<td>0.25</td>
<td>0.31</td>
<td>0.31</td>
<td>0.69</td>
<td>0.31</td>
<td>0.25</td>
</tr>
<tr>
<td>RTC</td>
<td>0.22</td>
<td>0.42</td>
<td>0.22</td>
<td>0.57</td>
<td>0.85</td>
<td>1.00</td>
<td>0.39</td>
<td>0.33</td>
<td>0.20</td>
<td>0.76</td>
<td>0.84</td>
<td>0.83</td>
<td>0.22</td>
<td>0.65</td>
<td>0.65</td>
<td>0.48</td>
<td>0.65</td>
<td>0.17</td>
</tr>
</tbody>
</table>

The graph of Circuit Task Complexity versus Robot Task Complexity graphically reveals the plotted differences in the researcher’s (in the role of instructor or teacher) expected level of complexity (i.e. the Circuit Task Complexity) and the students’ achievement (i.e. the Robot Task Complexity). Ideally one would expect the two plotted areas to merge; in other words, the researcher (or teacher) provides a task commensurate with the expected successful outcome developed by the learners. Looking at the graph in Fig. 7 this assumption mostly appears to be the case. However, we can also numerically represent the differences between anticipated task complexity and successful accomplishment. This is called Task Fidelity, and is calculated as:

Task Fidelity = Circuit Task Complexity - Robot Task Complexity

TF = CTC – RTC
Table IV and Fig. 8 portray the results of Task Fidelity plotted against the order of tasks of increasing challenge.

<table>
<thead>
<tr>
<th>TASK</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>T5</th>
<th>T6</th>
<th>T7</th>
<th>T8</th>
<th>T9</th>
<th>T10</th>
<th>T11</th>
<th>T12</th>
<th>T16</th>
<th>T17</th>
<th>T18</th>
<th>T19</th>
<th>T20</th>
<th>T21</th>
<th>T28</th>
</tr>
</thead>
<tbody>
<tr>
<td>TF</td>
<td>0.34</td>
<td>0.08</td>
<td>0.59</td>
<td>0.24</td>
<td>0.15</td>
<td>-0.31</td>
<td>-0.14</td>
<td>-0.02</td>
<td>-0.01</td>
<td>-0.13</td>
<td>-0.21</td>
<td>-0.27</td>
<td>0.03</td>
<td>-0.34</td>
<td>-0.34</td>
<td>0.21</td>
<td>-0.34</td>
<td>0.08</td>
</tr>
</tbody>
</table>

To reiterate, Task Fidelity is an indicator of the complexity of the circuit compared with the complexity of the program to complete the circuit. The zero line indicates ideal Task Fidelity; or ideal task complexity. Data plotted above the zero line indicate that the robot program was more complex than the circuit the robot had to maneuver. Data below the zero line indicate that the circuit was more complex than the robot program required to successfully navigate it. Our data reveal that for most tasks the programming required to complete the circuits was less than the considered complexity of the circuit (i.e. most data points are below the zero line of Task Fidelity). This appears to be the case across the range of challenges indicated by the students. The exceptions are T2, T4, T5 and T20 where TF values are above the zero line of Task Fidelity. Looking at Table I it appears that these tasks involved sensors. Programming of sensors is indeed more complex for the students and this was also reflected in the immersion data mentioned above; students were most anxious when engaged in tasks requiring sensor programming and were thus less immersed in the challenge.

However, as their skills of sensor programming increased, immersivity increased as indicated by Task 28 where Japanese students were taught by UK students to program the robot to follow a line (use of light sensor), stop (use of color sensor), and throw a ball. TF value for T28 was only +0.08; slightly above the optimal line.

The challenge for instructors is to seek tasks similar to T28 where immersivity is close to or on the optimal path of immersivity, and task complexity is close to or on the optimal line of Task Fidelity. The results indicate that over the period of this research experiment (one year) that researchers and students achieved an optimal learning task (i.e. T28). The next challenge is to seek ways to transfer these observations to further tasks with different participants in order to develop more reliable optimal learning tasks when collaborating in the programming of robots in a virtual space.
VIII. FUTURE WORK

We are adapting our virtual research space to support simulations of hazardous actions in and around a virtual representation of the Fukushima plant and surrounding structures such as a nuclear reactor; see Fig. 9. A number of telerobotic tasks (i.e. tasks T22 to T27) have been undertaken. In these tasks virtual controllers have been used to maneuver the real-world LEGO robot. The tasks have involved ‘search and rescue’ operations around various obstacles representing the Fukushima plant. An iPhone has been attached to the robot for streaming video live to the operator, while another camera has provided an aerial view of the robot’s movements. The robot has also been maneuvered into a dark space thereby necessitating the use of the flashlight on the iPhone while simultaneously streaming the video; the Bambuser App was used. These telerobotic tasks further aim to develop effective collaboration of Japanese and UK students, while researchers seek tasks with high immersivity, optimal task fidelity, and engaged learning.

As the project matures we would like children and adults to gain hands-on experience moving around our simulated virtual Fukushima space and observe a physical robot simultaneously moving at a remote location (e.g. in a scaled mock-up of a hazardous building such as the reactor). It is anticipated that such user-accessible simulations with citizens controlling the virtual robot will create an awareness and understanding of disaster recovery, and not simply rely upon retrospective information from unprepared experts [20]. As the context for the research will be located in a virtual nuclear power plant, our research will also support the ‘Japan Recovery’ initiative. As well as capturing data for analysis of cognitive processes, we also aim to familiarize students and the public with the complexities of nuclear power; given that there is much confusion about the situation at present here in Japan.

IX. CONCLUSION

This applied research is developing metrics for learning when conducting virtual world tasks. The motivation to implement this research was the nuclear disaster of 3-11. A virtual Fukushima nuclear plant and an OpenSim training space have been iteratively designed and built. International collaboration by students as non-experts has highlighted the benefits and challenges posed when engaged in constructing robot-mediated interactions (RMI) within the context of distance-based communication in 3D spaces. Students’ immersion (or flow), Circuit Task Complexity, and Robot Task Complexity have been calculated. Optimal learning tasks have been highlighted. A new metric is suggested for measuring tasks involving robots, which we term Task Fidelity.

ACKNOWLEDGMENTS

The research has been supported by the UK Prime Minister’s Initiative PMI2, Japan’s JAIST, and Future University’s special research funding. The author expresses thanks to UK collaborators and students at University of Glamorgan and Cynon Valley schools, research assistants at Future University, Japan, and metaverse designers at Reaction Grid USA.
this research was presented at the Human-Robot Interaction conference in Tokyo, Japan, March 3-6, 2013 and at the Experiential Learning in Virtual Worlds conference in Lisbon, Portugal, March 7-9, 2013.

REFERENCES


Michael Vallance Ed.D, MSc. is a Professor in the Department of Media Architecture, Future University Hakodate, Japan. He has been involved in educational technology design, implementation, research and consultancy for over fifteen years, working closely with Higher Education Institutes, schools and media companies in UK, Singapore, Malaysia and Japan. He was awarded second place in the Distributed Learning category of the United States Army’s 2012 Federal Virtual Worlds Challenge for his research in virtual collaboration. His website is http://www.mvallance.net.

Catherine Naamani, MA, FHEA is Head of Technology Enhanced Learning based in the Centre for Excellence in Learning & Teaching at the University of South Wales (formerly Glamorgan), South Wales, UK. She has worked with local schools for the past 6 years to promote the use of technology to enhance learning and teaching. During this time, she was also seconded to the Welsh Government to advise on developing an ICT Strategy for schools. In her current role, she leads the University’s strategy to research and develop innovative approaches to Technology Enhanced Learning, Teaching and Assessment. Her website is http://staff.southwales.ac.uk/users/1338-cnaamani.

Malcolm Thomas is a Mathematics teacher at Mountain Ash Comprehensive School, South Wales, UK.

James Thomas is a Physics teacher at Aberdare High School, South Wales, UK.