Poster: DejaVu: Visual Diffing of Cyber Physical Systems

Fergus Leahy
Imperial College London, UK
fergus.leahy@imperial.ac.uk

Naranker Dulay
Imperial College London, UK
n.dulay@imperial.ac.uk

ABSTRACT

In this abstract we present DejaVu, a 3D virtual world co-simulator for 'visual diffing' of cyber-physical system deployments in indoor and outdoor environments. Using faster-than-real-time simulation and efficient recording DejaVu can record days of simulation data, including environmental, sensor and network data for later replay and analysis. DejaVu enables developers to replay and visually compare multiple simulations simultaneously using different visual diffing techniques, including ghosts, paths, colour and size, highlighting differences between runs, including energy consumption, radio metrics, movement, etc. We demonstrate several of these visual diffing techniques in an CPS-enhanced evacuation case study.

CCS CONCEPTS

• **Computer systems organization** → **Embedded and cyber-physical systems**; • **Computing methodologies** → *Simulation environments*;

KEYWORDS

Cyber Physical Systems, Simulation, Analysis, 3D Virtual World

ACM Reference Format:
Fergus Leahy and Naranker Dulay. 2017. Poster: DejaVu: Visual Diffing of Cyber Physical Systems. In *Proceedings of BuildSys '17: The 4th ACM International Conference on Systems for Energy-Efficient Built Environments* . ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3137133.3141443

1 INTRODUCTION

Cyber-physical systems and IoT devices are becoming increasingly integrated into buildings around us, providing building climate-control, fire detection, security, lighting and numerous other services for homes and offices. Building and integrating these systems quickly becomes complex, with many different types of devices, vendor's ecosystems and protocols to deal with [3]. Real-world end-to-end testing of such systems, to ensure correctness, robustness and energy-sustainability, is time-consuming, expensive and often impractical or inconvenient, e.g., requiring repeated access to offices and corridors to set-up, test and move networks of nodes.

Simulation tools provide part of the solution to these issues, enabling efficient and scalable software testing on virtualised target devices, whilst remaining accurate enough for a successful deployment in the real-world [7, 9]. However, traditional test-beds and simulation tools have limited facilities for simulating the environment, simulation recording, playback and analysis. A typical simulation utilises scripted inputs or traces to provide input to running tests, recording all outputs and events to log files. Developers can observe simulated behaviour visually (through network maps, timelines and visual sensor feedback, such as LEDs) whilst it is running. After which, the recorded data can be viewed and compared to other runs textually, or through ad-hoc processing into visual graphs and plots etc. Although simulation runs can be recorded into video, this only provides a single fixed viewpoint to observe the simulation from. Thus, the tools and techniques for supporting developers test and analyse cyber-physical systems intuitively in simulation remains limited.

Modern video game engines are becoming a viable research tool due to an increase in computing power, supporting realistic graphics, improved physics and a more complex and interactive game world. This has led to a variety of research focusing on utilising them including CPS, autonomous driving and robotics [1, 2, 4–6, 8].

In this abstract we present the concept of visual diffing of CPS simulations, discuss and demonstrate how our co-simulator DejaVu, implements some of these features in our CPS-enhanced evacuation case study.

2 VISUAL DIFFING OF CYBER PHYSICAL SYSTEMS

Inspired by its use in other domains, such as live sports and video games, visual diffing enables viewers to see the differences between two or more instances of an event visually incorporated directly into the visual medium itself, instead of just presenting a simple data metric alongside, such as time. One technique is visual ghosting, e.g., two individual skiers racing, are visually overlaid as ghosts onto the same race track, provides a visually appealing and intuitive experience as the race progresses[1]. Similarly in DejaVu, instead of viewing simulations in isolation, developers can visual diff two simulations using ghosts, as shown in figure 1a, to show how differently crowds of people move.

2.1 DejaVu - Visual Diffing CPS Simulator

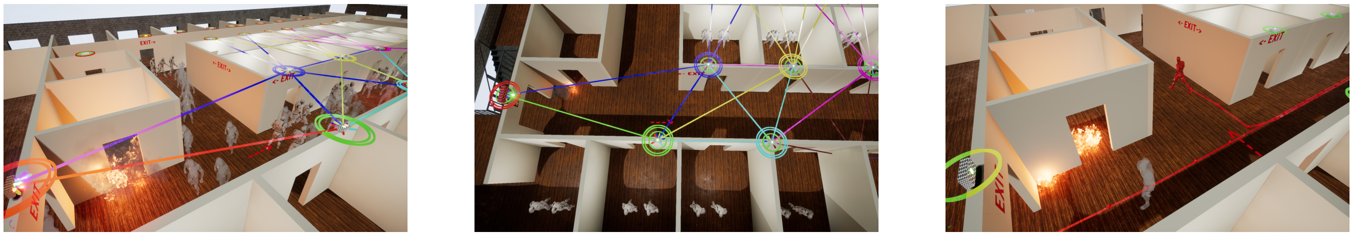
DejaVu is a 3D co-simulator platform for CPS and WSN, designed for running and comparing multiple simulation runs of a system in parallel. DejuVu is built using Cooja, a multi-level WSN simulator for Contiki-based nodes, and Unreal Engine 4 (UE4), a 3D game engine, providing a virtual world to virtually deploy real code and simulate the interactions between a CPS, the environment and virtual crowds. DejaVu can be used in both indoor and outdoor settings.

In order to support the visual diffing described in section 2.2, DejaVu utilises a time segmented compression scheme to efficiently

BuildSys '17: The 4th ACM International Conference on Systems for Energy-Efficient Built Environments , November 8–9, 2017, Delft, Netherlands

ACM ISBN 978-1-4503-5544-5/17/11.
https://doi.org/10.1145/3137133.3141443

[1] Dartfish SimulCam Ski race with ghosts: https://youtu.be/wZBOJnJM1XA



(a) Ghost: CPS-enhanced evacuation diffed with non-enhanced ghosts.

(b) Colour & size: A node on the left enlarged and red to alert a developer.

(c) Path: Evacuee paths in different runs based on CPS assistance.

Figure 1: Visual Diffing features of DejaVu

record days of simulation data, including data from the virtual environment, sensors and radio network, enabling a full reconstruction of simulation runs for later playback, review and analysis. Unlike video recordings, using this reconstruction technique, developers are able to move through not only time but also space within recorded simulations, stopping, rewinding and fast-forwarding to review recorded simulations from any point of view at any point in time.

2.2 Visual Diffing Techniques

For real world deployments, we are limited to what we can see through visual interfaces, external physical outputs or textual data logs. Similarly, by simply looking at the cyber-physical system in situ it is very difficult to observe activity and interaction between nodes. However, in the virtual world we can enhance our view of it, super-imposing or overlaying information which is otherwise hidden from view, such as radio traffic and sensor readings.

Taking this one step further, using a recording saved earlier, the simulator can simultaneously run one or more simulations in parallel, enabling observers to visually compare or diff them directly. However, the difficulty arises in how to represent these differences, including non-visual data, in intuitive ways.

DejaVu supports several visual diffing techniques including: ghosts, showing differences in location of people or objects, represented as translucent, ghastly coloured copies of themselves (figure 1a); paths, showing different paths people and objects take across many runs (figure 1c); colour, colour gradients and size, alerting and showing viewers at a glance the status and/or scale of difference between node meta-data, such as sensor readings, battery consumption or radio characteristics (figure 1b).

2.3 Case Study and Performance

To demonstrate DejaVu's features we developed a case study which focuses on the use of a cyber-physical enhanced fire detection and evacuation system for large buildings with multiple corridors and exit routes, such as offices or shopping malls. Utilising Cooja, we simulated a network of nodes running a distributed fire avoidance navigation algorithm. Evacuees observe directions signs controlled by nodes. Utilising DejaVu we were able to quickly iteratively develop and test the algorithm, observing how nodes performed and people reacted. When testing DejaVu, an evacuation simulation of an office floor with 30 sensors deployed and 60 evacuees, is capable of running at up to 3x real-time speed. DejaVu simulation recordings achieve significant savings when compared to raw recordings, resulting in sizes of 1min:40MB/1.6MB, 1hr:1.5GB/45MB, 24hr:36GB/1.1GB, for raw and compressed recordings respectively.

3 CONCLUSION

In this abstract we presented DejaVu, a 3D co-simulator supporting visual diffing of live and recorded CPS deployment simulations. We demonstrated the feasibility and capabilities that visual diffing provides through the use of an evacuation case study. DejaVu enables developers to cross-analyse simulated runs, highlighting points of interest and differences between runs with full time-control, seamlessly within the simulation itself.

REFERENCES

[1] C. E. Aguero and N. Koenig et al. 2015. Inside the Virtual Robotics Challenge: Simulating Real-Time Robotic Disaster Response. *IEEE Transactions on Automation Science and Engineering* 12, 2 (April 2015), 494–506. https://doi.org/10.1109/TASE.2014.2368997

[2] S. Carpin, M. Lewis, Jijun Wang, S. Balakirsky, and C. Scrapper. 2007. USARSim: a robot simulator for research and education. In *Robotics and Automation, 2007 IEEE International Conference on*. 1400–1405. https://doi.org/10.1109/ROBOT.2007.363180

[3] Stephen Dawson-Haggerty, Andrew Krioukov, Jay Taneja, Sagar Karandikar, Gabe Fierro, Nikita Kitaev, and David E Culler. 2013. BOSS: Building Operating System Services.. In *NSDI*, Vol. 13. 443–458.

[4] B. Kim, A. Jarandikar, J. Shum, S. Shiraishi, and M. Yamaura. 2016. The SMT-based automatic road network generation in vehicle simulation environment. In *2016 International Conference on Embedded Software (EMSOFT)*. 1–10. https://doi.org/10.1145/2968478.2968498

[5] N. Koenig and A. Howard. 2004. Design and use paradigms for Gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, Vol. 3. 2149–2154 vol.3. https://doi.org/10.1109/IROS.2004.1389727

[6] Fergus Leahy and Naranker Dulay. 2017. Ardán: Using 3D Game Engines in Cyber-Physical Simulations (Tool Paper). In *Cyber Physical Systems. Design, Modeling, and Evaluation: 6th International Workshop, CyPhy 2016, Pittsburgh, PA, USA, October 6, 2016*. Springer International Publishing, 61–70. https://doi.org/10.1007/978-3-319-51738-4_5

[7] Philip Levis, Nelson Lee, Matt Welsh, and David Culler. 2003. TOSSIM: Accurate and Scalable Simulation of Entire TinyOS Applications. In *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*. ACM, New York, NY, USA, 126–137. https://doi.org/10.1145/958491.958506

[8] W. Mueller, M. Becker, A. Elfeky, and A. DiPasquale. 2012. Virtual prototyping of Cyber-Physical Systems. In *Design Automation Conference (ASP-DAC), 2012 17th Asia and South Pacific*. 219–226. https://doi.org/10.1109/ASPDAC.2012.6164948

[9] F. Österlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt. 2006. Cross-Level Sensor Network Simulation with COOJA. In *Local Computer Networks, Proceedings 2006 31st IEEE Conference on*. 641–648. https://doi.org/10.1109/LCN.2006.322172