On Incorporating Search-Based Heuristics into Real-World Systems

Erik M. Fredericks
Oakland University
Rochester, MI, USA
fredericks@oakland.edu

Kate M. Bowers
Oakland University
Rochester, MI, USA
kmlabell@oakland.edu

Reihaneh H. Hariri
Oakland University
Rochester, MI, USA
rhosseinzadehha@oakland.edu

***Abstract*—While massive strides have been made in the field of search-based software testing (SBST) in recent years, there yet remains the problem of transitioning such techniques to reality. This paper discusses this problem in terms of cyber-physical systems, presents research challenges for applying SBST to this domain, and outlines the state-of-the-art achievements of the SBST community in this regard.**

***Index Terms*—real-world testing, cyber-physical systems, search-based software testing**

I. Introduction

Search-based software testing (SBST) relies on heuristics that often require a massive amount of resources (e.g., time and computing resources) to adequately explore the search space of feasible solutions for the testing problem at hand, including test case prioritization, test oracle generation, and mutant generation. SBST techniques generally require many evaluations on a system under test to ensure that an optimal solution is found. As such, SBST is often performed at design time, when computing resources are generally not a concern. The field of cyber-physical systems (CPS), however, must cope with real-world issues that can significantly impact the execution of the system [1]. CPSs are often described as the intersection between computational, physical, and network-based processes (i.e., significantly complicated systems). Such systems can include autonomous vehicles, smart cities, and robotics. As such, the complexity of scale with respect to verifying and validating CPS behaviors is enormous, thereby proving suitable for search-based testing heuristics.

In this paper, we discuss how the CPS domain, illustrated by a real-world CPS robotics application, is an ideal environment for SBST resulting from the numerous amounts of uncertainty (e.g., unexpected weather conditions, unanticipated parameter interactions, etc.) that can impact a system. We also outline key challenges to overcome and highlight existing research that can be applied to this domain.

II. Motivating Example

To illustrate a sample CPS, consider an autonomous humanoid robot (NASA's Valkyrie robot) as depicted in Figure 1. Valkyrie is intended to be deployed in extreme space environments (e.g., on the surface of Mars), providing NASA with valuable extraterrestrial data and possibly working with human counterparts in space [2], [3]. This robotic system exemplifies a CPS in that real-world concerns, such as concurrency, strict timing requirements, interaction with the environment, and communications are all first-class concerns. For example, performing autonomous pick-and-place operations with a robotic hand requires precision in control, communication with sensors, and an awareness of the surrounding environment.

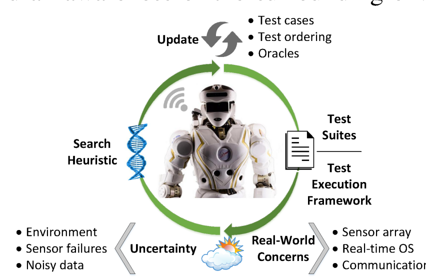

Fig. 1. NASA Valkyrie (R5) Robot [3] with run-time SBST cycle.

Testing a CPS. Figure 1 also demonstrates how SBST might be applied to the CPS domain. Assuming that a test suite (comprising test plans, test cases, oracles, etc.) has been defined for the system and that a run-time testing framework is implemented (e.g., continuous multi-agent testing [4], run-time self-adaptive testing [5], etc.), we anticipate a continuous feedback loop of performing tests at run time, interaction of real-world concerns and uncertainty (impacting test case results), a search heuristic (e.g., evolutionary search, multi-objective optimization, etc.) updating the test artifact, and then re-application to the testing framework.

III. SBST Challenges for Cyber-Physical Systems

The following challenges, at *minimum*, exist for performing SBST in the CPS domain:

Challenge (1) – Impact to behavior/performance. Executing test cases at run time on a CPS can impact its execution in terms of its expressed behaviors or the speed and/or reliability of its actions. For example, executing a test suite in parallel may successfully validate some behavior, however the time required to execute the test suite may conflict with the timing constraints of its real-time operating system (RTOS), thereby violating system constraints and causing a robotic hand to

This research has been supported in part by NSF grant CNS-1657061, the Michigan Space Grant Consortium, the Comcast Innovation Fund, and Oakland University. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of Oakland University or other research sponsors.

collide with a fragile sample. Moreover, CPSs are often *safety-critical* in that failure can result in damage to humans or the system itself, significant loss of monetary value, and damage to property/objects [6]. As such, any failures in this domain can be catastrophic (i.e., violation of safety properties/invariants) and must be avoided at all costs, test-induced or otherwise.

Challenge (2) – Uncertainty. Uncertainty can manifest in many forms with many definitions [7], however the common result is a system that misbehaves resulting from unanticipated or unaccounted for system and environmental conditions. Such conditions may have been expected but instantiated in a different manner (*known-unknown*: e.g., sensor expected to malfunction but to a harsher degree) or may not have been expected at all (*unknown-unknown*: e.g., a set of unseasonable weather conditions that "blinds" Valkyrie's sensor array). SBST provides an ideal strategy for minimizing uncertainty, as the search-based nature of this domain can be used to explore many different situations in which a CPS may exist. For instance, SBST could be used to generate as many instantiations of a test suite to fully exercise a CPS, even under conditions that were not accounted for in its requirements specification. However, the time required to search must also address Challenge (1).

Challenge (3) – Cyber-Physical Concerns. CPSs present an interesting opportunity resulting from the cyber-physical boundary. For instance, CPSs can be limited in terms of available on-board memory and/or processing power, where performing an exhaustive search of the solution space using available resources can very easily overexert the processor and over-utilize memory/storage. Incorporating SBST techniques, whether at design time or run time, would require an additional agent that not only can perform the search procedure, but also respect any timing requirements imposed by the RTOS (i.e., Challenge (1)) and minimize the amount of uncertainty surrounding the CPS (i.e., Challenge (2)).

An additional CPS concern lies in the transition to reality from simulation (i.e., the reality gap). Performing verification and validation techniques on a simulation of a system, in reality, only provides assurance for the simulation, not the real system itself (no matter how exhaustive verification/testing was performed). There will always be some measure of uncertainty that was unaccounted for in the simulation design, and as a result, systems absolutely require real-world testing, where we argue that SBST can provide an ideal suite of heuristics for reducing uncertainty and enhancing run-time assurance.

IV. Existing Research

Nguyen *et al*. previously introduced a continuous multi-agent testing framework, where separate agents validated other agents under test. The authors recently extended this concept with an evolutionary testing approach for deriving the fitness functions of autonomous agents [8]. We anticipate that an agent-based approach, whether in terms of localized- or cloud-based agents, can offset the impacts of Challenge (1) (i.e., impact to the system). However, techniques must be implemented in parallel to ensure that system behaviors and/or requirements do not present violations or exhibit less optimal outputs. Mansoor *et al*. recently performed lightweight verification and testing of surgical robots, a highly safety-critical domain [9]. González *et al*. described a model-based testing framework for performing CPS testing early in development [10]. We agree that a strenuous early testing phase can mitigate a large number of problems that may be encountered at run time. The combination of lightweight online verification and validation techniques, combined with a strong early testing phase, provides a promising approach to continuous run-time SBST in the CPS domain.

V. Discussion

This paper has discussed the challenges stemming from performing SBST within the CPS domain, including uncertainty, run-time concerns, and safety. While strides have been made by multiple communities for verifying and validating CPS projects, there still exists a large space for SBST to explore how to optimize testing techniques, whether at design time or run time. We encourage the SBST community to explore the CPS domain to demonstrate real-world usability, reproducibility, and effectiveness of SBST techniques.

Acknowledgment

The authors gratefully acknowledge the support of the members of the RIVeR research lab at Northeastern University.

References

[1] E. A. Lee, "Cyber physical systems: Design challenges," in *2008 11th IEEE Int'l. Symposium on Object and Component-Oriented Real-Time Distributed Computing*. IEEE, 2008, pp. 363–369.

[2] X. Long, P. Long, and T. Padir, "Compositional autonomy for humanoid robots with risk-aware decision-making," in *2017 IEEE-RAS 17th Int'l. Conf. on Humanoid Robotics (Humanoids)*. IEEE, 2017, pp. 553–560.

[3] NASA. (2015) Nasa awards two robots to university groups for r&d upgrades. [Online]. Available: https://www.nasa.gov/press-release/nasa-awards-two-robots-to-university-groups-for-rd-upgrades

[4] C. D. Nguyen, A. Perini, P. Tonella, and F. B. Kessler, "Automated continuous testing of multiagent systems," in *The 5th European Workshop on Multi-Agent Systems*, 2007.

[5] E. M. Fredericks, A. J. Ramirez, and B. H. C. Cheng, "Towards run-time testing of dynamic adaptive systems," in *Proceedings of the 8th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*, ser. SEAMS '13. IEEE Press, 2013, pp. 169–174.

[6] S. Mitra, T. Wongpiromsarn, and R. Murray, "Verifying cyber-physical interactions in safety-critical systems," *IEEE Security Privacy*, vol. 11, no. 4, pp. 28–37, 2013.

[7] M. Zhang, B. Selic, S. Ali, T. Yue, O. Okariz, and R. Norgren, "Understanding uncertainty in cyber-physical systems: a conceptual model," in *European conf. on modelling foundations and applications*. Springer, 2016, pp. 247–264.

[8] C. D. Nguyen, S. Miles, A. Perini, P. Tonella, M. Harman, and M. Luck, "Evolutionary testing of autonomous software agents," *Autonomous Agents and Multi-Agent Systems*, vol. 25, no. 2, pp. 260–283, 2012.

[9] N. Mansoor, J. A. Saddler, B. Silva, H. Bagheri, M. B. Cohen, and S. Farritor, "Modeling and testing a family of surgical robots: an experience report," in *Proc. of the 2018 26th ACM Joint Meeting on European Software Engineering Conf. and Symposium on the Foundations of Software Engineering*. ACM, 2018, pp. 785–790.

[10] C. A. González, M. Varmazyar, S. Nejati, L. C. Briand, and Y. Isasi, "Enabling model testing of cyber-physical systems," in *Proceedings of the 21th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems*, ser. MODELS '18. ACM, 2018, pp. 176–186.