Sensing Everywhere: Towards Safer and More Reliable Sensor-enabled Devices

Marta Kwiatkowska

Department of Computer Science, University of Oxford, Wolfson Building, Parks Road, Oxford OX1 3QD, UK

Abstract. In this age of ubiquitous computing we are witnessing ever increasing dependence on sensing technologies. Sensor-enabled smart devices are used in a broad range of applications, from environmental monitoring, where the main purpose is information gathering and appropriate response, through smartphones capable of autonomous function and localisation, to integrated and sometimes invasive control of physical processes. The latter group includes, for example, self-parking and self-driving cars, as well as implantable devices such as glucose monitors and cardiac pacemakers [1, 2]. Future potential developments in this area are endless, with nanotechnology and molecular sensing devices already envisaged [3].

These trends have naturally prompted a surge of interest in methodologies for ensuring safety and reliability of sensor-based devices. Device recalls [4] have added another dimension of safety concerns, leading FDA to tighten its oversight of medical devices. In seeking safety and reliability assurance, developers employ techniques to answer to queries such as “the smartphone will never disclose the bank account PIN number to unauthorised parties”, “the blood glucose level returns to a normal range in at most 3 hours” and “the probability of failure to raise alarm if the levels of airborne pollutant are unacceptably high is tolerably low”. Model-based design and automated verification technologies offer a number of advantages, particularly with regard to embedded software controllers: they enable rigorous software engineering methods such as automated verification in addition to testing, and have the potential to reduce the development effort through code generation and software reuse via product lines.

Automated verification has made great progress in recent years, resulting in a variety of software tools now integrated within software development environments. Models can be extracted from high-level design notations or even source code, represented as finite-state abstractions, and systematically analysed to establish if, e.g., the executions never violate a given temporal logic property. In cases where the focus is on safety, reliability and performance, it is necessary to include in the models quantitative aspects such as probability, time and energy usage. The preferred technique here is quantitative verification [5], which employs variants of Markov chains, annotated with reward structures, as models and aims establish quantitative properties, for example, calculating the probability or expectation of a given event. Tools such as the probabilistic model checker PRISM [6] are widely used to analyse safety, dependability and performability of system models in several application domains, including communication protocols, sensor networks and biological systems.

The lecture will give an overview of current research directions in automated verification for sensor-enabled devices. This will include software verification for TinyOS [7], aimed at improving the reliability of embedded software written in nesC; as well as analysis of sensor network protocols for collective decision making, where the increased levels of autonomy demand a stochastic games approach [8]. We will outline the promise and future challenges of the methods, including emerging applications at the molecular level [9] that are already attracting attention from the software engineering community [10].

Acknowledgement. This research has been supported in part by ERC grant VERIWARE and Oxford Martin School.

References

1. Sankaranarayanan, S., Fainekos, G.: Simulating insulin infusion pump risks by in-silico modeling of the insulin-glucose regulatory system. In: Proc. CMSB'12. LNCS, Springer (2012) To appear.
2. Jiang, Z., Pajic, M., Moarref, S., Alur, R., Mangharam, R.: Modeling and verification of a dual chamber implantable pacemaker. In: TACAS. (2012) 188–203
3. Kroeker, K.L.: The rise of molecular machines. Commun. ACM **54**(12) (2011) 11–13
4. Food, U., Drug Admin.: List of Device Recalls
5. Kwiatkowska, M.: Quantitative verification: Models, techniques and tools. In: Proc. 6th joint meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering (ESEC/FSE), ACM Press (September 2007) 449–458
6. Kwiatkowska, M., Norman, G., Parker, D.: PRISM 4.0: Verification of probabilistic real-time systems. In Gopalakrishnan, G., Qadeer, S., eds.: Proc. 23rd International Conference on Computer Aided Verification (CAV'11). Volume 6806 of LNCS., Springer (2011) 585–591
7. Bucur, D., Kwiatkowska, M.: On software verification for TinyOS. Journal of Software and Systems **84**(10) (2011) 1693–1707
8. Chen, T., Forejt, V., Kwiatkowska, M., Parker, D., Simaitis, A.: Automatic verification of competitive stochastic systems. In: Proc. 18th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'12). Volume 7214 of LNCS., Springer (2012) 315–330
9. Lakin, M., Parker, D., Cardelli, L., Kwiatkowska, M., Phillips, A.: Design and analysis of DNA strand displacement devices using probabilistic model checking. Journal of the Royal Society Interface **9**(72) (2012) 1470–1485
10. Lutz, R.R., Lutz, J.H., Lathrop, J.I., Klinge, T., Henderson, E., Mathur, D., Sheasha, D.A.: Engineering and verifying requirements for programmable self-assembling nanomachines. In: ICSE, IEEE (2012) 1361–1364